

Titre: Conception et analyse de la stabilité robuste d'une commande non linéaire d'un quadrotor
Title:

Auteur: Afaf Ziti
Author:

Date: 2019

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Ziti, A. (2019). Conception et analyse de la stabilité robuste d'une commande non linéaire d'un quadrotor [Mémoire de maîtrise, Polytechnique Montréal]. PolyPublie.
Citation: <https://publications.polymtl.ca/3881/>

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/3881/>
PolyPublie URL:

Directeurs de recherche: Lahcen Saydy, & Ouassima Akhrif
Advisors:

Programme: génie électrique
Program:

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Conception et analyse de la stabilité robuste d'une commande non linéaire d'un quadrotor

AFAF ZITI

Département de génie électrique

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*

Génie électrique

Avril 2019

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Ce mémoire intitulé:

Conception et analyse de la stabilité robuste d'une commande non linéaire d'un quadrotor

présenté par **Afaf ZITI**

en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*

a été dûment accepté par le jury d'examen constitué de :

Richard GOURDEAU, président

Lahcen SAYDY, membre et directeur de recherche

Ouassima AKHRIF, membre et codirecteur de recherche

Lyne WOODWARD, membre

REMERCIEMENTS

Au terme de ce travail, je tiens à remercier en premier lieu mon directeur de recherche M. Lahcen Saydy et ma co-directrice de recherche Mme Ouassima Akhrif, pour m'avoir offert l'opportunité d'accomplir ce travail, de m'avoir encadrée et soutenue techniquement, financièrement ainsi que moralement tout au long du projet. Leurs disponibilités et judicieux conseils ont contribué à la réalisation de ce mémoire.

Je tiens également à remercier les membres du jury, Mme Lyne Woodward et M. Richard Gourdeau pour le temps qu'ils ont consacré à évaluer mon mémoire de maîtrise et d'avoir accepté de participer à sa présentation.

Enfin, je tiens à témoigner ma reconnaissance envers ma famille et mes amis, pour m'avoir tout le temps encouragée et épaulée.

À tous ces intervenants, je présente mes remerciements, mon respect et ma gratitude.

RÉSUMÉ

Les performances d'une loi de commande basée sur le modèle dynamique d'un quadrotor sont susceptibles de se dégrader lorsque les paramètres du modèle sont incertains ou variables. L'objectif de ce projet de maîtrise est de remédier à une problématique industrielle exprimée par la compagnie ARA robotique. Il consiste à concevoir un contrôleur non linéaire à architecture fixe capable de permettre un suivi de trajectoires, respectant des spécifications précises, pour une gamme de quadrotors dont la masse varie à l'intérieur d'un intervalle $[m_1, m_2]$. Une loi de commande basée sur le backstepping avec action intégrale est développée dans ce cadre, le choix de ses gains est effectué par le biais de l'outil des applications gardiennes afin de garantir les spécifications du cahier de charge pour une variation de la masse du véhicule dans l'intervalle précité.

Pour ce faire, le modèle du système est obtenu grâce aux équations de Newton-Euler, ensuite le contrôleur non linéaire par backstepping avec action intégrale est conçu. Parallèlement, un contrôleur PID et un contrôleur par backstepping sont présentés pour servir de comparaison à la fin de ce mémoire.

La validation du contrôleur proposé est effectuée en simulation selon plusieurs scénarios représentant des environnements différents. Les résultats de simulation des diverses approches de commande sont comparés en se basant sur le critère de la moyenne quadratique de l'erreur. Ainsi, la moindre moyenne est obtenue pour l'algorithme de commande par backstepping intégral. Ce dernier, tel qu'il est développé, est capable de fonctionner sur l'éventail de modèles de quadrotors d'intérêt.

ABSTRACT

The performance of a controller based on a dynamical model of a quadrotor can be significantly deteriorated when the model parameters are uncertain or variable. The purpose of this project is to tackle an industrial problem expressed by the company ARA robotique pertaining to this issue. The aim of this thesis is to design a fixed-structure nonlinear trajectory tracking controller, respecting given specifications, for a range of quadrotor models with varying masses in a given interval $[m_1, m_2]$. In this context, a control law based on integral backstepping is developed. Its gains are tuned using the tool of guardian maps in order to guarantee the specifications for a variation of the vehicle mass within the above-mentioned range.

In order to achieve this, a dynamical model is obtained employing the Newton-Euler formalism, then a nonlinear controller based on integral backstepping is developed. In parallel, a PID controller and a backstepping controller are presented. They are used for comparison purposes at the end of this thesis.

The validation of the proposed controller is performed by simulation according to multiple scenarios representing different environments. The simulation results of the various control approaches are compared applying the mean squared error criterion. Thus, the least value is obtained when the integral backstepping control algorithm is used. This controller, as it is developed, can operate on the set of quadrotors of interest.

TABLE DES MATIÈRES

REMERCIEMENTS	III
RÉSUMÉ.....	IV
ABSTRACT	V
TABLE DES MATIÈRES	VI
LISTE DES TABLEAUX.....	IX
LISTE DES FIGURES.....	X
LISTE DES SIGLES ET ABRÉVIATIONS	XIII
LISTE DES ANNEXES.....	XV
CHAPITRE 1 INTRODUCTION.....	1
CHAPITRE 2 REVUE DE LITTÉRATURE	4
2.1 Les contrôleurs linéaires.....	4
2.2 Les contrôleurs non linéaires.....	7
CHAPITRE 3 NOTIONS PRÉLIMINAIRES	10
3.1 Préliminaires.....	10
3.2 Théorie de la commande backstepping	11
3.3 Algorithme de conception	11
CHAPITRE 4 MODÉLISATION DU QUADROTOR.....	17
4.1 Repères	17
4.2 Effets physiques agissant sur le quadrotor	17
4.2.1 Les forces	18
4.2.2 Les moments	19
4.3 Le modèle du quadrotor selon Newton-Euler	21
4.4 Le modèle linéaire du système	23

4.5	Relation entre les forces/moments et les vitesses des moteurs	24
CHAPITRE 5 CONCEPTION DES CONTRÔLEURS		26
5.1	Contrôle linéaire du quadrotor	26
5.2	Contrôle du quadrotor par backstepping	30
5.3	Contrôle du quadrotor par backstepping avec action intégrale	31
5.3.1	Contrôle du système de positionnement.....	32
5.3.2	Contrôle du système d'orientation	35
5.3.3	Le système en boucle fermée	37
5.3.4	Signaux de commande	38
CHAPITRE 6 APPLICATIONS GARDIENNES		40
6.1	Problématique de robustesse	40
6.2	Présentation théorique	40
6.2.1	Définitions et propriétés	41
6.2.2	Applications gardiennes usuelles	42
6.2.3	Exemple d'application.....	44
6.3	Application au cas du quadrotor.....	47
6.3.1	Modèle incertain.....	47
6.3.2	Stratégie d'analyse	48
6.3.3	Vérification de la stabilité pour des paramètres nominaux	50
6.3.4	Calcul de l'application gardienne.....	51
6.3.5	Choix des gains du contrôleur.....	56
6.4	Remarques.....	60
CHAPITRE 7 SIMULATIONS DES CONTRÔLEURS ET VALIDATION		63
7.1	Modèle de simulation	63

7.1.1	Modèle du vent.....	64
7.1.2	Génération de la trajectoire	65
7.1.3	Les paramètres de simulations	66
7.2	Les scénarios de simulations	67
7.2.1	Scénario 1: Environnement idéal	67
7.2.2	Scénario 2: Perturbations provenant du vent.....	68
7.2.3	Scénario 3: Incertitude paramétrique	68
7.3	Simulation du contrôleur PID	69
7.3.1	Scénario 1	70
7.3.2	Scénario 2.....	73
7.3.3	Scénario 3.....	75
7.4	Simulation du contrôleur Backstepping	76
7.4.1	Scénario 1	76
7.4.2	Scénario 2.....	79
7.4.3	Scénario 3.....	80
7.5	Simulation du contrôleur backstepping avec action intégrale.....	82
7.5.1	Scénario 1	82
7.5.2	Scénario 2.....	84
7.5.3	Scénario 3.....	86
7.6	Comparaison des résultats.....	87
CHAPITRE 8	CONCLUSION ET PERSPECTIVES	90
BIBLIOGRAPHIE	92
ANNEXES	95

LISTE DES TABLEAUX

Tableau 5.1: Valeurs propres de la matrice A aux points (r_1, r_2)	46
Tableau 5.2: Gains nominaux.....	51
Tableau 5.3: Les contours à évaluer.....	55
Tableau 5.4: Les gains utilisés lors d'une itération de calcul.....	57
Tableau 5.5: Valeurs des gains retenus	58
Tableau 5.6: Les modèles de simulation	59
Tableau 6.1: Paramètres du quadrotor.....	66
Tableau 6.2: Paramètres du quadrotor (suite)	67
Tableau 6.3: Points de la trajectoire 1	68
Tableau 6.4: Points de la trajectoire 2	68
Tableau 6.5: Gains du contrôleur PID.....	70
Tableau 6.6: Comparaison des performances	71
Tableau 6.7: Gains du contrôleur backstepping	77
Tableau 6.8: Les gains du contrôleur par backstepping avec action intégrale.....	82
Tableau 6.9: Comparaison dans le cas du scénario idéal	88
Tableau 6.10: Comparaison en présence du vent.....	88
Tableau 6.11: Comparaison en présence d'incertitudes.....	89

LISTE DES FIGURES

Figure 3.1: Schématisation des forces de poussée [30].....	19
Figure 4.1: Les boucles de contrôle du quadrotor	27
Figure 4.2: Contrôleur PID.....	28
Figure 4.3: Le mouvement latéral du quadrotor.....	28
Figure 4.4: Le mouvement longitudinal du quadrotor	29
Figure 5.1: Domaines de stabilité usuels.....	44
Figure 5.2: Points d'annulation de v_α et v_ξ	46
Figure 5.3: Le domaine Ω de stabilité généralisée	49
Figure 5.4: Représentation de $v_{\alpha 1} = 0$ dans l'espace $(\gamma_2, \gamma_3, \gamma_1)$	52
Figure 5.5: Représentation de $v_{\alpha 2} = 0$ dans l'espace $(\gamma_2, \gamma_3, \gamma_1)$	53
Figure 5.6: Représentation de $v_\xi = 0$ dans l'espace $(\gamma_2, \gamma_3, \gamma_1)$	53
Figure 5.7: Représentation de v dans l'espace $(\gamma_2, \gamma_3, \gamma_1)$	54
Figure 5.8: Découpe suivant γ_1 et intersection des composantes stables (la zone orange)	55
Figure 5.9: Les valeurs propres de A_{BF} selon les contours	55
Figure 5.10: Évolution de la composante stable suivant des valeurs de γ_1	56
Figure 5.11: Points d'annulation de v pour $\gamma_1 = 400$	57
Figure 5.12: Simulation avec les gains retenus	59
Figure 6.1: Structure du système de simulation	63
Figure 6.2: Modèle de simulation sous Matlab/Simulink	64
Figure 6.3: Exemple de vent généré en utilisant le modèle de Dryden.....	65
Figure 6.4: Exemple d'une trajectoire à accélération constante.....	66
Figure 6.5: Stabilisation autour de x_d, y_d, z_d, ψ_d : comparaison entre le modèle linéarisé et le modèle non linéaire	71

Figure 6.6: Suivi de la trajectoire désirée dans l'espace, cas du PID	72
Figure 6.7: Suivi de trajectoires selon les six degrés de liberté, cas du PID.....	72
Figure 6.8: Vitesses de rotation des moteurs, cas du PID	73
Figure 6.9: La force du vent, cas du PID	74
Figure 6.10: Suivi de trajectoires en présence du vent, cas du PID	74
Figure 6.11: Suivi de trajectoires en présence d'incertitudes, cas du PID.....	75
Figure 6.12: Erreurs de suivi en présence d'incertitudes, cas du PID	76
Figure 6.13: Suivi de la trajectoire désirée dans l'espace, cas du backstepping.....	77
Figure 6.14: Suivi de trajectoires selon les six degrés de liberté, cas du backstepping	78
Figure 6.15: Vitesses de rotation des moteurs, cas du backstepping	78
Figure 6.16: La force du vent, cas du backstepping.....	79
Figure 6.17: Suivi de trajectoires en présence du vent, cas du backstepping	80
Figure 6.18: Suivi de trajectoires en présence d'incertitudes, cas du backstepping.....	81
Figure 6.19: Erreurs de suivi en présence d'incertitudes, cas du backstepping	81
Figure 6.20: Suivi de la trajectoire désirée dans l'espace, cas du backstepping avec action intégrale	83
Figure 6.21: Suivi de trajectoires selon les six degrés de liberté, cas du backstepping avec action intégrale	83
Figure 6.22: Vitesses de rotation des moteurs, cas du backstepping avec action intégrale	84
Figure 6.23: La force du vent, cas du backstepping avec action intégrale.....	85
Figure 6.24: Suivi de trajectoires en présence du vent, cas du backstepping avec action intégrale	85
Figure 6.25: Suivi de trajectoires en présence d'incertitudes, cas du backstepping avec action intégrale.....	86

Figure 6.26: Erreurs de suivi en présence d'incertitudes, cas du backstepping avec action intégrale	87
Figure 6.27: Les angles d'Euler	95
Figure 6.28: Singularité de la matrice ρ , tirée de [40].....	97

LISTE DES SIGLES ET ABRÉVIATIONS

UAV	Unmanened Aerial Vehicule
LQR	Linear Quadratic Regulator
PID	Proportionnel Intégral Dérivé
NED	North East Down
ISS	Input to State Stability
SISO	Single Input and Single Output

Notations:

\mathcal{F}_i	Repère inertiel
\mathcal{F}_b	Repère du quadrotor
η	Vecteur de position exprimé dans \mathcal{F}_i (m)
η_d	Vecteur de position désirée exprimé dans \mathcal{F}_i (m)
V	Vitesse linéaire exprimée dans \mathcal{F}_b (m/s)
μ	Vitesse linéaire exprimée dans \mathcal{F}_i (m/s)
Θ	Vecteur des angles d'Euler (rad)
ϕ	Roulis (rad)
θ	Tangage (rad)
ψ	Lacet (rad)
ω	Vecteur des vitesses angulaires du repère \mathcal{F}_b par rapport au repère \mathcal{F}_i exprimé dans \mathcal{F}_b (rad/s)
T	Force de poussée (N)
τ	Vecteur des couples exprimé dans \mathcal{F}_b (Nm)

G_r	Vecteur de précession gyroscopique exprimé dans \mathcal{F}_b (Nm)
ρ	Matrice de propagation des angles d'Euler
R	Matrice de rotation de \mathcal{F}_b par rapport à \mathcal{F}_i
I	Matrice d'inertie du quadrotor exprimée dans \mathcal{F}_b ($kg.m^2$)
g	Accélération gravitationnelle (m/s^2)

LISTE DES ANNEXES

Annexe 1 - Mouvement d'un corps rigide à six degrés de liberté.....	95
---	----

CHAPITRE 1 INTRODUCTION

Les drones sont des engins volants autonomes ou semi-autonomes capables d'être autopilotés ou commandés à distance. Ils sont très souvent désignés par l'abréviation UAV qui réfère à « Unmanned Aerial Vehicles ».

L'intérêt porté à ces robots volants ne cesse de s'accroître. Bien que leur champ d'application ait été limité au début à des fins militaires, telles que la surveillance et la localisation des mines terrestres, leurs utilisations s'étendent aujourd'hui aux domaines civils comme la gestion des infrastructures (barrages, lignes à haute tension), les services météorologiques (mesure de la température et du taux d'humidité), la télécommunication (relais de télécommunication), ainsi que la photographie aérienne (presse, production des films). Ils ont rendu possible l'accomplissement de tâches dangereuses dans un environnement hostile sans risque de perte ou de menace humaine.

Il existe aujourd'hui une grande variété de drones dont la classification peut se faire selon plusieurs critères, entre autres la taille (longueur, envergure), la voilure (fixe, tournante, battante), l'altitude ou l'endurance. Dans ce mémoire, nous nous intéressons en particulier à un engin volant miniature et multi-rotors de type quadrotor et plus précisément à la conception d'une loi de commande lui permettant de suivre des trajectoires désignées par l'utilisateur. Ce véhicule présente certains défis. Non seulement il est sensible aux perturbations à cause du fort couplage entre les variables d'états et les entrées de commande, mais en plus sa dynamique est sous actionnée, c.-à-d. le nombre des entrées est inférieur au nombre des degrés de liberté. Par conséquent, la technique de contrôle à adopter doit surmonter ces obstacles.

De surcroît, la conception d'une loi de commande se base sur la connaissance du modèle du système, ses paramètres physiques en particulier. Une variation de ceux-ci est susceptible de dégrader la réponse du système en boucle fermée, voire entraîner une instabilité. Dans ces conditions, il serait nécessaire de reconfigurer le contrôleur afin de l'adapter au nouveau modèle qui tient compte de la modification des paramètres. Bien souvent, ceci nécessite un travail supplémentaire pour le recalcul des gains du contrôle. Cette problématique a suscité un intérêt industriel, de la part de ARA Robotique, pour le développement d'un contrôleur standard capable de contrôler plusieurs drones possédant la même configuration mais de masses différentes.

C'est dans ce cadre que ce projet de maîtrise a vu le jour. L'idée du projet est de synthétiser un contrôleur backstepping avec une action intégrale permettant le suivi de trajectoires à un drone quadrotor. Le choix de cette stratégie de contrôle permet la représentation par la suite du contrôleur en une forme PID. Cette configuration a permis l'obtention d'une forme linéaire du système en boucle fermée sur laquelle les outils linéaires d'analyse de la robustesse seront applicables, les applications gardiennes en particulier. En se basant sur cet outil, le contrôleur est conçu pour être fonctionnel sur un éventail de quadrotors.

L'objectif principal de ce travail concerne alors la conception d'un contrôleur permettant à une gamme de quadrotors de masses différentes d'auto naviguer selon une trajectoire sélectionnée. Ce contrôleur doit assurer les performances à la fois sur le modèle nominal et sur les modèles obtenus pour d'autres valeurs du paramètre incertain qui est la masse.

Dans un premier temps, la loi de commande est réalisée pour permettre une poursuite de la trajectoire au quadrotor avec des paramètres nominaux. Par la suite, la masse du véhicule est considérée comme un paramètre incertain nécessitant une analyse de la robustesse du contrôleur. À ce niveau, l'outil des applications gardiennes est appliqué pour un travail de synthèse avec performances robustes, à savoir le choix de gains du contrôleur à une architecture fixe conservant les performances pour un intervalle souhaité de la variation de la masse du quadrotor.

L'objectif secondaire est la comparaison du contrôleur backstepping muni d'une action intégrale avec un contrôleur backstepping classique et un contrôleur PID linéaire afin de montrer l'intérêt de la conception proposée.

Ce mémoire consiste en 6 chapitres.

Le premier chapitre est dédié à une revue de la littérature sur les méthodes de contrôle linéaire et non linéaire appliquées au quadrotor. Le deuxième chapitre fait l'objet d'introduction des notions concernant la théorie du backstepping, de même que les principaux résultats utiles au travail de la conception. Le troisième chapitre s'intéresse à la modélisation de la dynamique du quadrotor. Nous y présentons les équations cinématiques et dynamiques décrivant le fonctionnement du quadrotor. Le quatrième chapitre détaille la conception d'un contrôleur linéaire PID et d'un contrôleur backstepping avec action intégrale. Le cinquième chapitre présente une approche par applications gardiennes pour un choix robuste des gains du contrôleur backstepping avec action intégrale permettant le respect des spécifications désirées. Le sixième chapitre se consacre à la simulation

des différents contrôleurs selon trois scénarios représentant un environnement idéal, un environnement à perturbations ainsi qu'une présence d'incertitude paramétrique et aborde une comparaison des résultats. In fine, le dernier chapitre clôt ce mémoire avec une conclusion et des recommandations d'investigations futures à la lumière des résultats obtenus.

CHAPITRE 2 REVUE DE LITTÉRATURE

Le développement des méthodes de fabrication des drones, les avancements liés aux systèmes de navigation, la capacité de contrôle à distance ainsi que le progrès des technologies de stockage d'énergie ont influé l'évolution de la production de ces aéronefs et la diversité de leurs utilisations [1].

Les drones quadrotors ont pu être privilégiés pour des applications autonomes grâce à leur simplicité mécanique, leur capacité à opérer des vols stationnaires et à exécuter des décollages et atterrissages verticaux. Cette popularité a mené à une attention particulière envers ce véhicule. De nombreux travaux de recherche ont été effectués sur le sujet dans le but d'étudier la dynamique et le contrôle desdits drones. Ce chapitre est alors consacré à la présentation de quelques travaux axés sur le contrôle linéaire et non linéaire des quadrotors. Nous soulignons que la liste qui sera dressée est loin d'être exhaustive, elle nous révèle toutefois la diversité des stratégies pouvant être employées (voir les revues de littérature dans [2-4] pour d'autres travaux portant sur le contrôle des quadrotors).

2.1 Les contrôleurs linéaires

Les techniques de contrôle linéaire sont les premières utilisées pour la commande du vol des UAV. Elles incluent la commande par PID, LQR, H_∞ et les contrôleurs par séquençement de gain [5].

- La commande par PID

L'un des contrôleurs linéaires les plus utilisés est le proportionnel-intégral-dérivé (ou PID) grâce à la facilité d'ajustement des gains, et le degré de robustesse acceptable qu'il assure. Une structure hiérarchique constituée de deux boucles est généralement employée pour la commande des quadrotors. La boucle interne est composée de PID afin de contrôler l'orientation du véhicule. La boucle externe utilise quant à elle des PID découplés pour contrôler le mouvement de translation [3].

Bien que sa structure soit simple, le régulateur PID présente un défi. L'abondance de méthodes de réglage rend la détermination de ses gains une opération délicate. Il s'agit de trouver un compromis entre le dépassement, l'erreur en régime permanent, la stabilité et la robustesse.

Cette approche est examinée en [6] où les auteurs ont réussi à implanter ce type de contrôleur avec succès. L'objectif du travail est de stabiliser les angles d'orientation du quadrotor uniquement. Pour ce faire, l'auteur a simplifié la dynamique en négligeant les effets gyroscopiques ce qui a mené à éliminer les termes de couplage. Le contrôleur conçu reste toutefois incapable de stabiliser le système en présence de perturbations importantes.

La même architecture a été sélectionnée par [7]. La position et l'orientation sont à contrôler toutes les deux dans ce travail. Les simulations ainsi que les tests expérimentaux ont permis de démontrer les performances du contrôleur. Un faible dépassement, une erreur statique quasi nulle et une réponse rapide sont obtenus.

Malgré ces réalisations, des défis majeurs subsistent pour ce type de contrôleur, en l'occurrence, la non-linéarité du modèle mathématique et son imprécision en présence de dynamiques non modélisées. De surcroît, le réglage des gains du contrôleur est effectué autour d'un point de fonctionnement. Ces raisons font que le contrôle du quadrotor par PID limite ses performances.

- La commande linéaire quadratique

La commande linéaire quadratique LQR (Linear Quadratic Regulator) est un type de commande optimale qui permet de déterminer la commande d'un système minimisant un critère de performance quadratique dont la fonction coût se met généralement sous la forme :

$$J = \int_0^{\infty} (X^T Q X + U^T R U + X^T N U) dt \quad (2-1)$$

où X et U représentent respectivement le vecteur d'état du système et le vecteur d'entrée; Q, R et N sont des matrices de pondération à régler. En résolvant une équation de Riccati, un gain de retour d'état dit optimal K est trouvé et la commande $U = KX$ permet de stabiliser le système en minimisant le critère (2-1).

Cette technique est utilisée dans [6], mais les performances du contrôleur testé expérimentalement n'ont pas été satisfaisants. L'erreur en régime permanent ne s'y annule pas et des oscillations apparaissent. L'auteur l'explique par les imperfections du modèle dû à la non-considération de la dynamique des moteurs particulièrement. Le travail cité en [8] a, par la suite, considéré le modèle dynamique complet dans le but de faire suivre au drone une trajectoire verticale prédéfinie. La conception du contrôleur est faite sur la base du modèle linéarisé. Le contrôleur résultant est ensuite

implanté sur le modèle non linéaire pour les simulations. Les résultats présentés ont démontré l'efficacité du contrôleur dans un environnement parfait et en présence de perturbations dues à un vent de 0.1 m/sec suivant l'axe des x , cette valeur reste tout de même faible par rapport à une vitesse de vent à l'extérieur en une journée normale. La même technique de commande est expérimentée dans un travail plus récent [9], dont l'objectif est de faire suivre à un quadrotor Quanser Qball-X4 une trajectoire horizontale fixée au préalable, le contrôleur LQR y est comparé à un contrôleur PID, les résultats de la simulation et du vol dans le laboratoire montrent des performances meilleures du contrôleur LQR.

- La commande H_∞

La commande H_∞ est la commande robuste la plus utilisée, elle assure performance et rejet de perturbations ainsi que garantie théorique de la robustesse de la stabilité du système. La méthode se base sur la traduction des exigences fréquentielles et temporelles en filtres de pondération, de sorte que le problème devient un problème de minimisation de norme H_∞ . Nous citons le livre [10] pour les détails de la théorie.

Cette approche a l'inconvénient de fournir un contrôleur d'ordre plus élevé que le système à asservir, la méthode H_∞ structurée vient remédier à ce problème en donnant la possibilité de concevoir un contrôleur avec une architecture fixée au préalable [11].

L'approche H_∞ est explorée dans une panoplie de travaux de recherche dont nous mentionnons [12-14]. Le travail cité en [12] compare la méthode H_∞ à une méthode de contrôle basée sur trois boucles différentes dont une est responsable de compenser les incertitudes paramétriques et les perturbations, les simulations et l'expérimentation qui concernaient uniquement le vol stationnaire ont montré que le contrôleur linéaire basé sur H_∞ est capable de rejeter les perturbations, mais demeure limité dans la réduction des effets non linéaires lorsqu'implanté sur le système réel. Le même constat est relevé dans [14] où le contrôleur perd sa capacité de suivi de trajectoire d'altitude, lorsqu'implanté sur le système non linéaire complet.

Ces stratégies de commande précitées sont simples à implanter et présentent des résultats satisfaisants, mais leurs performances ne sont garanties qu'en un voisinage du point de fonctionnement autour duquel la linéarisation est effectuée. Pour ces raisons, des algorithmes de commande dont le domaine de fonctionnement est plus large doivent être développés.

2.2 Les contrôleurs non linéaires

▪ Linéarisation entrée-sortie

Le principe de la commande par linéarisation entrée-sortie est de trouver une dynamique linéaire équivalente du système non linéaire en effectuant un changement de variables en choisissant une commande adéquate. Pour ce faire, la sortie est dérivée plusieurs fois par rapport au temps jusqu'à l'apparition de l'entrée de commande. L'expression de cette dernière est par la suite choisie afin d'éliminer les non-linéarités et garantir la stabilité du système ou la poursuite de trajectoires de référence. Cette méthode nécessite que le modèle soit parfaitement connu, chose qui n'est pas facilement assurable en réalité. D'autre part si le système a un degré relatif inférieur à l'ordre du système, il devient nécessaire d'étudier la stabilité de la dynamique interne, à savoir, la dynamique des zéros. Si cette dynamique est instable, la commande linéarisante ne pourra pas garantir la stabilité du système global.

La commande des quadrotors par linéarisation entrée-sortie a fait l'objet de plusieurs travaux avec différentes approches. Dans [15] le système est transformé en sous-systèmes SISO auxquelles la linéarisation entrée-sortie est appliquée, les simulations ont validé la stabilité et la robustesse de la loi de contrôle en présence de vent (force de vent de 10 N) et d'incertitudes paramétriques (20 % d'incertitudes sur les moments d'inertie). Une approche différente est adoptée dans [16] où l'hypothèse des petits angles est supposée afin de simplifier le modèle du système. Par la suite, seules les sorties x, y, z sont considérées pour déterminer les entrées de commande par linéarisation entrée sortie. Quant à l'angle ψ , il est commandé par un contrôleur PD. Dans le même esprit de réduction de complexité du système, [17] a divisé le contrôle du système global en deux parties : la première pour contrôler les angles d'orientation du quadrotor et la deuxième pour asservir sa position. Les résultats de simulation ont prouvé l'efficacité de la conception pour de faibles vitesses de référence.

▪ La commande par modes glissants

Le contrôle par modes glissants est articulé sur l'application d'un signal de commande non continu (à commutation) pour forcer le système à atteindre une surface appelée "surface de glissement", sélectionnée en fonction des objectifs de la commande, et d'y rester par la suite (glisser). Cette

technique n'omet pas les non-linéarités du système, ce qui constitue l'un de ses principaux avantages conjointement avec le bon suivi de trajectoire qu'elle assure [2].

Cette technique est appliquée dans [18] sur la dynamique de l'orientation du quadrotor ce qui a permis de stabiliser les angles d'Euler. Cependant, la présence de l'effet de réticence (shattering effect) était remarquable malgré les efforts déployés pour le réduire. Cette problématique est l'essence du travail [19]. L'auteur de cet article a comparé quatre types de contrôleurs basés sur le mode glissant de deuxième ordre afin de déduire celui qui élimine les effets non désirables de la commutation. Bien que les résultats de simulation et expérimentaux aient montré un bon suivi de trajectoire et un rejet de perturbations, ils demeurent limités puisque les contrôleurs se sont restreints à asservir l'altitude du drone indépendamment du reste de sa dynamique. Cette stratégie de commande a été jumelée à d'autres méthodes non linéaires, notamment le backstepping et le contrôle adaptatif, en vue d'augmenter la robustesse du système en présence d'incertitudes et perturbations externes [20, 21].

▪ Backstepping

Le backstepping est une méthode de commande non linéaire basée sur la théorie de stabilité de Lyapunov. Elle est bien adaptée aux systèmes sous actionnés. En effet, en réorganisant le système en sous-systèmes virtuellement en cascade [22], il devient possible de contrôler quelques états grâce à d'autres états qui agissent comme leurs entrées.

L'application du backstepping aux quadrotors a commencé au début des années 2000 avec [18]. La conception y est faite en trois étapes, d'abord le contrôle du système de la rotation puis le contrôle du déplacement vertical et enfin le contrôle du mouvement horizontal. Toutefois, seule la stabilisation du système de l'orientation a été testée sur un banc d'essai en maintenant le quadrotor à une altitude fixe. Les résultats ont montré que le contrôleur est capable de stabiliser le système considéré. Dans le cadre du même projet, une action intégrale est ajoutée en une étape ultérieure [23] afin de pouvoir rejeter les perturbations. Tous les états du système sont contrôlés cette fois-ci, par conséquent, le quadrotor a pu faire un décollage vertical et un mouvement rectangulaire à une hauteur faible avec succès. [24] présente la conception d'une commande par backstepping pour contrôler le modèle complet du quadrotor qui est composé de trois sous-systèmes interconnectés. Le premier est un sous-système sous actionné décrivant la dynamique suivant x et y en fonction des angles de roulis et de tangage, le deuxième sous-système complètement actionné

donnant la dynamique de l'altitude z et du lacet ψ et un dernier sous-système représentant la dynamique des rotors. La loi de commande que présente ce travail a été simulée pour un suivi de trajectoire tridimensionnelle. Alors que le test expérimental effectué concernait uniquement les mouvements verticaux et du lacet. Un autre projet à citer utilisant la commande par backstepping est présenté dans le mémoire [25]. La méthodologie de contrôle adoptée consiste aussi à décomposer la dynamique du drone en une dynamique de translation et une autre de rotation. La particularité de ce travail réside dans sa considération de l'effet de l'erreur de position angulaire sur la stabilité dynamique du quadrotor. Pour ce faire, la stabilité du système a été prouvée grâce à la théorie de stabilité de Lyapunov et à la théorie de stabilité entrée à états (ISS). D'autre part, les dérivées des commandes virtuelles y sont estimées plutôt que calculées directement et le contrôleur qui y est conçu garantit un parfait suivi de trajectoire dans l'espace en simulation, mais son fonctionnement expérimental reste limité en présence du vent.

La technique du backstepping a été mise en œuvre également dans le cas de défaillance d'une des hélices, en l'occurrence le travail en [26]. Ce dernier vise à faire atterrir le quadrotor en toute sécurité en cas d'urgence à l'aide d'un contrôleur par backstepping pour le mouvement de translation, associé à un contrôleur basé sur le PID pour le mouvement de rotation. Des simulations d'un vol en salle ont été effectuées pour confirmer la validité du contrôleur.

Tout au long de chapitre, nous avons exposé une panoplie de techniques de contrôle des véhicules aériens multi-rotors. L'adoption d'une technique se fait au gré de l'objectif de l'étude et du niveau de performance souhaitée.

Pour ce travail, la loi de commande du quadrotor sera réalisée à l'aide du backstepping avec action intégrale.

Le prochain chapitre passe en revue les notions théoriques sur lesquelles se basera la conception du contrôleur.

CHAPITRE 3 NOTIONS PRÉLIMINAIRES

La finalité de ce chapitre est d'exposer les principaux outils et définitions mathématiques servant, au développement de la loi de commande se basant sur la technique du backstepping. Nous y présentons les étapes de conception d'un contrôleur par backstepping et les théorèmes démontrant sa stabilité asymptotique.

La technique de commande par backstepping se base sur la théorie de stabilité de Lyapunov comme la majorité des techniques de commande non linéaires, elle propose une méthode systématique pour la conception d'un contrôleur pour un système non linéaire ([22]).

3.1 Préliminaires

Nous considérons dans ce paragraphe le système

$$\dot{x} = f(x) \quad (3-1)$$

admettant $x_e = 0$ pour point d'équilibre, c.-à-d. tel que $f(x_e) = 0$ et où f est localement Lipschitz ([22]).

Définition 3.1

Soit une fonction $V: \mathbb{R}^n \rightarrow \mathbb{R}$. V est dite :

- Définie positive si: $V(x) > 0, \forall x \neq 0$ et $V(0) = 0$
- Semi-définie positive si: $V(x) \geq 0, \forall x \neq 0$ et $V(0) = 0$
- Définie négative si: $V(x) < 0, \forall x \neq 0$ et $V(0) = 0$
- Semi-définie négative si: $V(x) \leq 0, \forall x \neq 0$ et $V(0) = 0$

Théorème 3.1: Théorème de Lyapunov (méthode directe)

Soit $D \subset \mathbb{R}^n$ un voisinage de x_e et $V(x): D \rightarrow \mathbb{R}$ une fonction continûment différentiable définie positive. Alors :

- $\dot{V} \leq 0 \forall x \in D \Rightarrow x_e$ est stable
- $\dot{V} < 0 \forall x \in D - \{0\} \Rightarrow x_e$ est asymptotiquement stable

si en plus $D = \mathbb{R}^n$ et $\dot{V}(x) \xrightarrow[||x|| \rightarrow \infty]{} \infty$ alors x_e est globalement asymptotiquement stable

3.2 Théorie de la commande backstepping

La conception par backstepping s'applique en général aux systèmes sous forme dite de rétroaction stricte (Strict-Feedback Form), c'est-à-dire ayant la forme en (3-2).

$$\begin{aligned}
 \dot{x}_1 &= f_1(x_1) + g_1(x_1)x_2 \\
 \dot{x}_2 &= f_2(x_1, x_2) + g_2(x_1, x_2)x_3 \\
 \dot{x}_3 &= f_3(x_1, x_2, x_3) + g_3(x_1, x_2, x_3)x_4 \\
 &\vdots \\
 \dot{x}_{n-1} &= f_{n-1}(x_1, x_2, \dots, x_{n-1}) + g_{n-1}(x_1, x_2, \dots, x_{n-1})x_n \\
 \dot{x}_n &= f_n(x_1, x_2, \dots, x_n) + g_n(x_1, x_2, \dots, x_n)u
 \end{aligned} \tag{3-2}$$

où $x = (x_1, x_2, \dots, x_n)^T \in \mathbb{R}^n$ est l'état du système et u est son entrée. Les f_i et g_i sont des fonctions non linéaires connues tel que $f_i(0) = 0$ et $g_i \neq 0$.

Il est à noter que le système qui fera l'objet de notre étude se présentera sous forme de deux sous-systèmes à rétroaction stricte.

L'objectif principal du backstepping est de trouver une fonction candidate de Lyapunov définie positive, dont la dérivée est définie négative assurant la stabilité asymptotique globale du système. Cette fonction est construite d'une manière récursive. En effet, le système global est divisé en n sous-systèmes en cascade. Chaque étape de conception k consiste à trouver une commande virtuelle afin d'assurer la stabilité du sous-système k en question. À l'étape n , la fin du processus, la commande réelle u du système étudié est obtenue. La succession des étapes de calcul permet également la formation de la fonction de Lyapunov du système global [22].

3.3 Algorithme de conception

Afin d'illustrer le principe de la méthode de backstepping, prenons le système (3-2) pour $n = 2$:

$$\begin{cases} \dot{x}_1 = f_1(x_1) + g_1(x_1)x_2 \\ \dot{x}_2 = f_2(x_1, x_2) + g_2(x_1, x_2)u \end{cases} \quad (3-3)$$

où g_1 et g_2 sont des fonctions non nulles.

Le problème de conception consiste à trouver une commande u qui stabilise le système globalement.

Sans perte de généralité, l'origine est considérée comme point d'équilibre du système pour une entrée nulle.

Étape 1:

On considère la première équation du système. Stabiliser ce sous-système est l'objectif de cette étape.

La sélection de la première fonction candidate de Lyapunov doit faire apparaître la dynamique de la variable d'état x_1 afin de pouvoir déterminer la première commande virtuelle (3-4). Une fonction quadratique constitue un bon choix de la fonction candidate de Lyapunov [27].

Soit

$$V_1(x_1) := \frac{1}{2}x_1^2 \quad (3-4)$$

Sa dérivée est donnée ci-dessous

$$\begin{aligned} \dot{V}_1(x_1) &= x_1 \dot{x}_1 \\ &= x_1(f_1(x_1) + g_1(x_1)x_2) \end{aligned} \quad (3-5)$$

À partir de (3-5) il est clair que si x_2 pouvait être choisi de façon judicieuse de telle sorte à avoir \dot{V}_1 négative, alors la stabilité de l'origine de la première équation en (3-3) serait assurée.

La variable d'état x_2 est traitée en cette étape comme une entrée et on définit sa valeur désirée comme étant :

$$x_{2d} := \alpha_1 \quad (3-6)$$

L'état x_2 n'est pas une entrée réelle du système, c'est pourquoi α_1 est nommée une commande virtuelle. Soit α_1 telle que:

$$f_1(x_1) + g_1(x_1)\alpha_1 = -\gamma_1 x_1 \quad (3-7)$$

où γ_1 est un paramètre strictement positif. Conséquemment, la commande virtuelle aura la valeur

$$\alpha_1 = \frac{1}{g_1(x_1)}(-\gamma_1 x_1 - f_1(x_1)) \quad (3-8)$$

Étape 2 :

Nous définissons la variable d'erreur e_1 , qui représente l'écart entre l'état x_2 et sa valeur désirée α_1 par:

$$e_1 := x_2 - \alpha_1 \quad (3-9)$$

La conception dans cette étape doit forcer l'écart e_1 à s'annuler avec une certaine dynamique.

La fonction candidate de Lyapunov est augmentée en V_2 :

$$V_2 = V_1 + \frac{1}{2}e_1^2 \quad (3-10)$$

V_2 a comme fonction dérivée :

$$\begin{aligned} \dot{V}_2 &= x\dot{x} + e_1\dot{e}_1 \\ &= -\gamma_1 x^2 + e_1(\dot{x}_2 - \dot{\alpha}_1) \\ &= -\gamma_1 x^2 + e_1(f_2(x_1, x_2) + g_2(x_1, x_2)u - \dot{\alpha}_1) \end{aligned} \quad (3-11)$$

La commande réelle est maintenant présente. Un choix pertinent de celle-ci est:

$$u = \frac{1}{g_2(x_1, x_2)}(\dot{\alpha}_1 - f_2(x_1, x_2) - \gamma_2 e_1) \quad (3-12)$$

où γ_2 est un paramètre strictement positif et $\dot{\alpha}_1$ est analytiquement calculée conformément à

$$\dot{\alpha}_1 = \frac{\partial \alpha_1}{\partial x_1} \dot{x}_1 \quad (3-13)$$

Avec un tel choix, on obtient :

$$\dot{V}_2 = -\gamma_1 x_1^2 - \gamma_2 e_1^2 \leq 0 \quad (3-14)$$

D'où la stabilité en boucle fermée du système original en (3-3).

Il est à noter que les paramètres γ_1 et γ_2 sont liés à l'emplacement des pôles de la matrice du système en boucle fermée (voir exemple ci-après), c.-à-d. le choix de ces deux paramètres fixe la dynamique de cette boucle.

Pour un système d'ordre $n \geq 2$, le processus est répété n fois jusqu'à l'obtention de la commande réelle u et de la fonction de Lyapunov V .

Exemple d'application

Soit le système sous forme rétroaction stricte ayant le modèle d'état non linéaire (3-15).

$$\begin{cases} \dot{x}_1 = -x_1^3 + x_2 \\ \dot{x}_2 = x_2^2 + u \end{cases} \quad (3-15)$$

On désire faire suivre à x_1 le signal de référence x_r , où \dot{x}_r et \ddot{x}_r sont supposées connues et suffisamment lisses.

Étape 1 :

La première étape concerne la première équation en (3-15). Définissons la variable d'écart :

$$e_1 := x_1 - x_r \quad (3-16)$$

Soit la candidate de Lyapunov $V_1(e_1) = \frac{1}{2} e_1^2$.

Sa dérivée temporelle est:

$$\dot{V}_1 = e_1 \dot{e}_1 = e_1 (-x_1^3 + x_2 - \dot{x}_r) \quad (3-17)$$

En choisissant la commande virtuelle $\alpha_1 = -k_1 e_1 + x_1^3 + \dot{x}_r$, alors si x_2 tend vers α_1 la dérivée de V_1 devient :

$$\dot{V}_1 = -k_1 e_1^2 \quad (3-18)$$

Toutefois, nous ne pouvons pas forcer x_2 à être égal à α_1 puisqu'il n'est pas une entrée directe du système, mais nous pouvons contrôler la variation de x_2 à partir de la deuxième équation d'état.

Étape 2 :

Définissons maintenant l'écart e_2 entre la variable d'état x_2 et sa valeur désirée α_1 .

$$e_2 := x_2 - \alpha_1 \quad (3-19)$$

Puis augmentons la candidate de Lyapunov en $V = V_1 + \frac{1}{2} e_2^2$.

Sa dérivée temporelle est donnée par :

$$\begin{aligned} \dot{V} &= e_1 \dot{e}_1 + e_2 \dot{e}_2 \\ &= -k_1 e_1^2 + e_2 (e_1 + x_2^2 + u - k_1 (e_2 - k_1 e_1) + 3x_1^5 - 3x_1^2 x_2 - \ddot{x}_r) \end{aligned} \quad (3-20)$$

À présent, la commande réelle du système figure sur la dernière équation. Il suffit de choisir la valeur de u pour rendre négative cette dérivée. Un tel choix est donné par :

$$u = -k_2 e_2 - e_1 - x_2^2 + k_1 (e_2 - k_1 e_1) - 3x_1^5 + 3x_1^2 x_2 + \ddot{x}_r \quad (3-21)$$

où k_2 est strictement positif, la dérivée s'écrit alors

$$\dot{V} = -k_1 e_1^2 - k_2 e_2^2 \leq 0 \quad (3-22)$$

Ce qui implique que la sortie $y = x_1$ suivra la référence x_r avec une certaine dynamique.

Le système en boucle fermée s'exprime sous la forme suivante :

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \end{bmatrix} = \begin{bmatrix} -k_1 & 1 \\ -1 & -k_2 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} \quad (3-23)$$

Il est intéressant de noter que la vitesse de convergence de x_1 vers le signal de référence dépend de la dynamique de la matrice en (3-23), en d'autres termes, les valeurs des gains k_1 et k_2 dépendent de la position désirée des pôles en boucle fermée.

Les étapes de conception d'un contrôleur basé sur la théorie du backstepping ont été présentées dans cette section, les notions théoriques relatives à la démonstration de la stabilité y figurent également. Il est à noter qu'un contrôleur backstepping ne garantit pas des erreurs nulles en présence de perturbations externes. Évidemment ceci est dû à l'absence de l'action intégrale nécessaire pour l'élimination des erreurs résiduelles. Pour y remédier, une action intégrale sera ajoutée virtuellement au système lors de la conception donnant lieu à un contrôleur backstepping avec action intégrale.

CHAPITRE 4 MODÉLISATION DU QUADROTOR

Il est primordial de comprendre la dynamique du système à étudier avant d'aborder la conception des lois de commande. À cette fin, ce chapitre traite de la modélisation du drone multi-rotor du type quadrotor. Dans un premier temps, nous procéderons à l'identification et à la représentation dans les repères adéquats des effets agissant sur le quadrotor. Dans un second temps, nous nous servirons des équations mécaniques d'un corps rigide selon le formalisme de Newton-Euler afin de développer le modèle dynamique du véhicule.

À la fin de chapitre, un modèle non linéaire est mis en évidence aussi bien qu'un modèle linéaire obtenu par linéarisation autour du point d'équilibre.

4.1 Repères

Il est nécessaire de définir deux repères pour pouvoir décrire le mouvement du quadrotor.

Le premier repère est le repère inertiel. Il s'agit d'un repère galiléen dans lequel les lois de Newton sont applicables. En aéronautique, le repère inertiel utilisé est le repère NED (North-East-Down). Ce repère sera noté $\mathcal{F}_i = (O_i, \mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i)$, où \mathbf{x}_i pointe vers le nord géographique, \mathbf{y}_i vers l'est et \mathbf{z}_i vers le centre de la Terre. Nous considérons que la surface de la Terre est localement plate, par conséquent \mathbf{x}_i et \mathbf{y}_i sont parallèle au sol et \mathbf{z}_i est perpendiculaire au sol.

Le deuxième repère est nécessaire à la description de l'orientation de l'aéronef. Il s'agit d'un référentiel lié au châssis du quadrotor et se déplace avec lui, il est identifié par $\mathcal{F}_b = (O_b, \mathbf{x}_b, \mathbf{y}_b, \mathbf{z}_b)$. L'origine O_b qui coïncide avec le centre de gravité, \mathbf{x}_b , \mathbf{y}_b et \mathbf{z}_b pointent respectivement vers l'avant, vers le bras droit et vers le bas.

4.2 Effets physiques agissant sur le quadrotor

Les forces et les moments considérés dans ce travail opérant sur le quadrotor sont :

- La force de poussée générée par chaque rotor
- La force gravitationnelle
- Les moments de traînée générés par la rotation
- Les moments gyroscopiques

Nous ne considérons pas les effets suivants :

- Les battements d'hélice
- La friction de l'air
- L'effet du sol
- L'instabilité de l'air et du vent

4.2.1 Les forces

La force de poussée

Elle résulte de la rotation des moteurs M_1, M_2, M_3, M_4 et des hélices et permet de soulever le quadrotor. Elle est perpendiculaire sur le plan des hélices (Figure 4.1).

La force de poussée générée par l'hélice i est:

$$T_i = C_T \rho A (\Omega_i r)^2 \quad (4-1)$$

où :

- C_T : terme aérodynamique dépendant de la géométrie de l'hélice
- ρ : la densité de l'air
- A : l'aire du disque généré par l'hélice, $A = \pi r^2$
- Ω_i : la vitesse de l'hélice
- r : la longueur de l'hélice

En supposant que l'aéronef effectue un vol à basse altitude avec une vitesse faible et que l'hélice est rigide, les variables précédentes peuvent alors être supposés constants [28, 29], et nous pouvons faire l'approximation suivante :

$$T_i = k_T \Omega_i^2 \quad (4-2)$$

où k_T est le coefficient de poussée.

La force de poussée totale, générée par les quatre moteurs, exprimée dans \mathcal{F}_b est :

$$F_{poussée} = \begin{bmatrix} 0 \\ 0 \\ -k_T (\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \end{bmatrix} \quad (4-3)$$

La force gravitationnelle

Cette force due au champ gravitationnel terrestre est exprimée dans le repère terrestre par :

$$F_g = m g \mathbf{z}_i \quad (4-4)$$

où g est l'accélération gravitationnelle approximée à 9.81 m/s^2 et m est la masse du quadrotor.

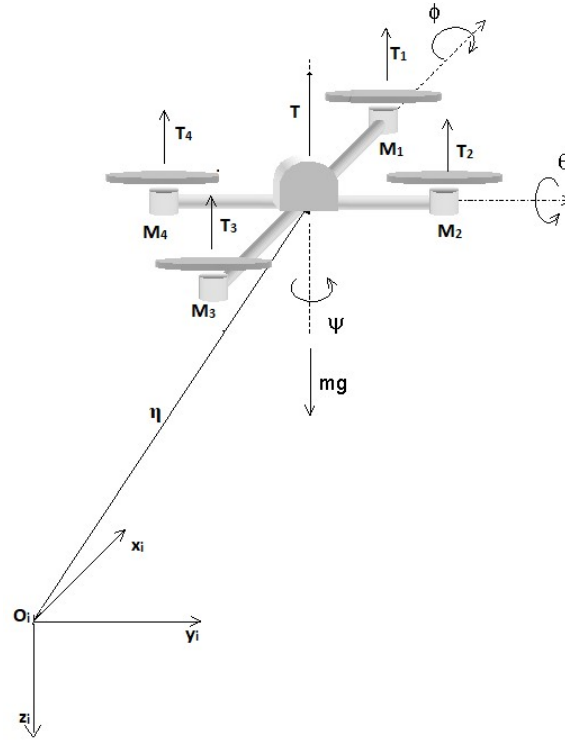


Figure 4.1: Schématisation des forces de poussée [30]

4.2.2 Les moments

Les moments dus aux forces de poussée

Ces moments sont créés par la différence entre les forces de portance de chaque paire de moteurs.

Elles sont exprimées ainsi :

$$\tau_{forces} = \begin{bmatrix} k_T l (\Omega_4^2 - \Omega_2^2) \\ k_T l (\Omega_1^2 - \Omega_3^2) \\ 0 \end{bmatrix} \quad (4-5)$$

k_t réfère au coefficient de poussée et l à la longueur de l'hélice

Le moment de traînée

Il est dû à l'angle d'attaque entre l'hélice et l'air. En effet, la rotation de l'hélice pousse l'air vers le bas pour générer la force de portance, néanmoins, l'angle d'attaque non nul donne naissance à une autre force perpendiculaire à la portance. Pour chaque hélice i , ce moment est donné par [29] :

$$D_i = C_D \rho A (\Omega_i r)^2 r^2 \quad (4-6)$$

où :

- C_D : terme aérodynamique dépendant de la géométrie de l'hélice et des conditions aéronautiques
- ρ : la densité de l'air
- A : l'aire du disque généré par l'hélice, $A = \pi r^2$
- Ω_i : la vitesse de l'hélice
- r : la longueur de l'hélice

Les mêmes hypothèses utilisées pour le calcul de la force de portance sont considérées ici, on obtient :

$$D_i = k_D \Omega_i^2 \quad (4-7)$$

où k_D est le coefficient de traînée. Dans cette équation, une rotation dans le sens horaire génère un moment négatif et une rotation dans le sens antihoraire génère un moment positif. Le moment résultant des quatre hélices est :

$$\tau_{traînée} = \begin{bmatrix} 0 \\ 0 \\ k_D(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2) \end{bmatrix} \quad (4-8)$$

Le moment gyroscopique: il correspond au phénomène de conservation d'axe de rotation d'un objet effectuant un mouvement de rotation. Si une force vient modifier l'axe de rotation, l'objet s'oppose à cette action en imposant un couple résistant à ce changement. Dans le cas du quadricoptère, il s'applique aux hélices qui ont leurs propres axes de rotation, mais qui sont perturbées par la rotation du rotor en entier. Le moment créé par chaque hélice est donné par :

$$G_i = \omega \times I_{helice} \Omega_i \mathbf{z}_i \quad (4-9)$$

où I_{hel} est l'inertie du rotor, $\omega = [p, q, r]^T$ est le vecteur de la vitesse angulaire et Ω_i la vitesse de rotation du moteur i .

Le moment gyroscopique total créé par les quatre rotors est :

$$G_r = \begin{bmatrix} I_{helice} q (-\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4) \\ -I_{hel} p (-\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4) \\ 0 \end{bmatrix} \quad (4-10)$$

On note par la suite $\Omega_r = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4$.

4.3 Le modèle du quadrotor selon Newton-Euler

Le modèle qui sera développé dans ce paragraphe suppose les hypothèses suivantes :

- Le quadrotor est un corps rigide
- La structure de l'aéronef est symétrique
- Le centre de gravité et l'origine du repère lié à l'objet sont superposés
- Les hélices sont rigides

Tenant compte de ces hypothèses, nous pouvons appliquer les équations du mouvement du corps rigide à six degrés de liberté présentées en Annexe A au quadrotor. En outre, à partir de la section 4.2, les forces et moments que subit le quadrotor exprimés dans le repère \mathcal{F}_b ont les expressions suivantes :

$$F = \begin{bmatrix} 0 \\ 0 \\ -k_T(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \end{bmatrix} + \begin{bmatrix} -mg \sin(\theta) \\ mg \sin(\phi) \cos(\theta) \\ mg \cos(\phi) \cos(\theta) \end{bmatrix} \quad (4-11)$$

$$\tau - G_r = \begin{bmatrix} k_T l (\Omega_4^2 - \Omega_2^2) \\ k_T l (\Omega_1^2 - \Omega_3^2) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ k_D (-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2) \end{bmatrix} - \begin{bmatrix} I_{helice} q \Omega_r \\ -I_{hel} p \Omega_r \\ 0 \end{bmatrix} \quad (4-12)$$

En appliquant les équations (8-13) et (8-16) de l'Annexe A, le modèle du quadrotor exprimé par rapport à \mathcal{F}_b est :

$$\begin{aligned}
\dot{\eta} &= RV \\
\dot{V} &= -\frac{1}{m}T\mathbf{k} + R^T g \cdot \mathbf{k} - [\omega]_{\times} V \\
\dot{\Theta} &= \rho(\Theta)\omega \\
I \dot{\omega} &= \tau - G_r - [\omega]_{\times} I \omega
\end{aligned} \tag{4-13}$$

où $\eta = [x, y, z]^T$ est la position du quadrotor exprimée dans \mathcal{F}_i , $V = [u, v, w]^T$ est sa vitesse linéaire exprimée dans \mathcal{F}_b , $\Theta = [\phi, \theta, \psi]^T$ est le vecteur des angles d'Euler, $\omega = [p, q, r]^T$ la vitesse angulaire exprimée dans \mathcal{F}_b , $\mathbf{k} = [0 \ 0 \ 1]^T$ et $\rho(\Theta)$ la matrice de transformation de $\dot{\Theta}$ à ω (voir (8-8) à l'Annexe A). $\rho(\Theta)$ est donnée par :

$$\rho(\Theta) = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \frac{\sin(\phi)}{\cos(\theta)} & \frac{\cos(\phi)}{\cos(\theta)} \end{bmatrix}$$

En notant μ la vitesse linéaire dans le repère inertiel, le même modèle exprimé dans le repère \mathcal{F}_i devient :

$$\begin{aligned}
\dot{\eta} &= \mu \\
\dot{\mu} &= -\frac{1}{m}TR\mathbf{k} + g \cdot \mathbf{k} \\
\dot{\Theta} &= \rho(\Theta)\omega \\
I \dot{\omega} &= \tau - G_r - [\omega]_{\times} I \omega
\end{aligned} \tag{4-14}$$

Un choix possible du vecteur d'état est $X = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \phi, \theta, \psi, p, q, r]^T$. Avec ce choix, on a la représentation d'état du modèle en (4-13):

$$\begin{cases}
 \ddot{x} = -\frac{1}{m}T(\cos(\phi)\sin(\theta)\cos(\psi) + \sin(\phi)\sin(\psi)) \\
 \ddot{y} = -\frac{1}{m}T(\cos(\phi)\sin(\theta)\sin(\psi) - \sin(\phi)\cos(\psi)) \\
 \ddot{z} = g - \frac{1}{m}T\cos(\phi)\cos(\theta) \\
 \dot{\phi} = p + \sin(\phi)\tan(\theta)q + \cos(\phi)\tan(\theta)r \\
 \dot{\theta} = \cos(\phi)q - \sin(\theta)r \\
 \dot{\psi} = \frac{\sin(\phi)}{\cos(\theta)}q + \frac{\cos(\phi)}{\cos(\theta)}r \\
 \dot{p} = \frac{1}{I_{xx}}[(I_{yy} - I_{zz})qr - I_{helic}q\Omega_r + \tau_\phi] \\
 \dot{q} = \frac{1}{I_{yy}}[(I_{zz} - I_{xx})rp + I_{helice}p\Omega_r + \tau_\theta] \\
 \dot{r} = \frac{1}{I_{zz}}[(I_{xx} - I_{yy})pq + \tau_\psi]
 \end{cases} \quad (4-15)$$

4.4 Le modèle linéaire du système

Le modèle présenté en (4-15) est linéarisé autour du point d'équilibre qui coïncide avec le point de fonctionnement « vol stationnaire ». Nous nous intéressons aux sorties qui décrivent la position du quadrotor dans l'espace ainsi que son orientation.

On considère le vecteur d'état $X = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \phi, \theta, \psi, p, q, r]^T$ et le vecteur de sorties $Y = [x, y, z, \phi, \theta, \psi]^T$, l'entrée du système étant $U = [T, \tau_\phi, \tau_\theta, \tau_\psi]^T$.

La linéarisation autour de l'équilibre $X_e = [x_e, y_e, z_e, 0, 0, 0, 0, 0, \psi_e, 0, 0, 0]^T$, $U_e = [mg, 0, 0, 0]^T$ donne le modèle d'état linéaire :

$$\begin{cases}
 \dot{X} = AX + BU \\
 Y = CX + DU
 \end{cases} \quad (4-16)$$

où :

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -g & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & g & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{1}{m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{1}{I_{xx}} & 0 & 0 \\ 0 & 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}; D = 0_{(6 \times 4)}$$

Rappelons que g est la constante gravitationnelle, I_{xx} , I_{yy} et I_{zz} sont les moments d'inertie axiaux.

4.5 Relation entre les forces/moments et les vitesses des moteurs

Le modèle développé précédemment possède comme entrée de commande la force de poussée T et les moments τ , tandis que les entrées réelles du quadrotor sont les vitesses de rotation des moteurs Ω_i . À partir des équations exposées auparavant, ces deux types d'entrées sont liés comme suit:

$$\begin{aligned}
T &= k_T(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \\
\tau_\phi &= k_T l(\Omega_4^2 - \Omega_2^2) \\
\tau_\theta &= k_T l(\Omega_1^2 - \Omega_3^2) \\
\tau_\psi &= k_D(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2)
\end{aligned} \tag{4-17}$$

Ces équations peuvent être réécrites sous forme matricielle telle qu'en (4-18) :

$$\begin{bmatrix} T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} k_t & k_t & k_t & k_t \\ 0 & -k_t l & 0 & k_t l \\ k_t l & 0 & -k_t l & 0 \\ -k_D & k_D & -k_D & k_D \end{bmatrix} \begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_2^2 \\ \Omega_2^2 \end{bmatrix} \quad (4-18)$$

La matrice en (4-18) est inversible, et son inverse permet d'obtenir la relation suivante :

$$\begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_2^2 \\ \Omega_2^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{4k_t} & 0 & \frac{1}{2k_t l} & -\frac{1}{4k_D} \\ \frac{1}{4k_t} & -\frac{1}{2k_t l} & 0 & \frac{1}{4k_D} \\ \frac{1}{4k_t} & 0 & -\frac{1}{2k_t l} & -\frac{1}{4k_D} \\ \frac{1}{4k_t} & \frac{1}{2k_t l} & 0 & \frac{1}{4k_D} \end{bmatrix} \begin{bmatrix} T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \quad (4-19)$$

Nous avons présenté dans ce chapitre les équations cinématiques et dynamiques dictant le mouvement du quadrotor dans l'espace, ces dernières soulignent la complexité du modèle, sa non-linéarité et le couplage entre ses différents états. Nous disposons maintenant d'un modèle dynamique sur lequel se basera la conception des contrôleurs.

CHAPITRE 5 CONCEPTION DES CONTRÔLEURS

Nous avons présenté dans les chapitres précédents le modèle mathématique et les outils théoriques nécessaires à la commande du quadrotor. Dans le présent chapitre, nous proposons le développement de lois de commande du vol. L'étude comporte trois volets: dans le premier, nous synthétisons un contrôleur linéaire de type PID basé sur le modèle linéaire développé précédemment, le deuxième volet sera consacré à la présentation d'un contrôleur par backstepping et nous finirons par le troisième volet où la conception du contrôleur par backstepping avec action intégrale est effectuée.

5.1 Contrôle linéaire du quadrotor

Dans l'industrie, la plupart des contrôleurs utilisés sont des contrôleurs PID. En effet, ils sont caractérisés par leur simplicité et leur performance.

Ce type de régulateur se base sur la génération d'un signal de commande à partir de l'écart e entre la référence et la mesure. Conformément à ça, la loi de commande est donnée par:

$$u(t) = k_p e(t) + k_d \frac{de(t)}{dt} + k_i \int_0^t e(\tau) d\tau \quad (5-1)$$

où e représente l'écart $y_{ref} - y$, y_{ref} le signal de référence et y la sortie mesurée et où k_p est le gain proportionnel, k_d est le gain dérivé et k_i est le gain intégral.

Le modèle du quadrotor considéré dans ce paragraphe est le modèle linéaire développé en (4-16).

Le contrôle du quadrotor est réalisé par une structure en cascade constituée de deux boucles. Le contrôleur de la boucle interne liée à la dynamique d'orientation est chargé de suivre la référence d'orientation du drone, c.-à-d. suivre les angles ϕ_d , θ_d et ψ_d et la boucle externe liée à la dynamique de position qui génère ces angles de roulis et de tangage de référence ainsi que la force de poussée totale nécessaires au positionnement du drone à un endroit de l'espace comme illustré à la Figure 5.1.

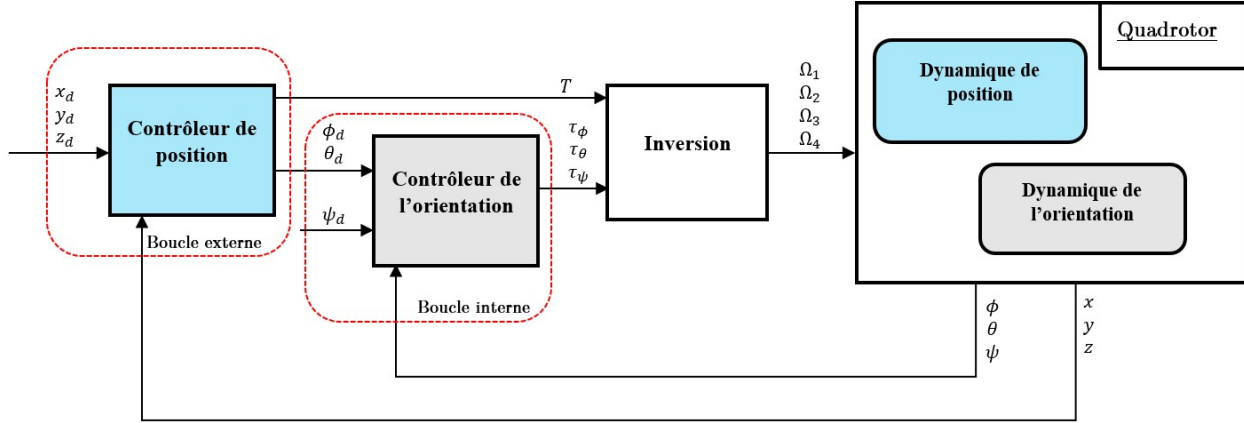


Figure 5.1: Les boucles de contrôle du quadrotor

En général, la stabilisation d'un quadrotor est réalisable par un contrôleur PD pour chaque degré de liberté. Cependant, une erreur résiduelle est non annulée au niveau de l'altitude z lorsque le quadrotor est en vol stationnaire à cause de l'effet de son poids. Pour cette raison, un contrôleur PD est conçu pour chaque degré de liberté à l'exclusion de l'altitude pour laquelle un contrôleur PID a été choisi. Les erreurs de suivi relatives à ceux-ci sont définies en (5-2), où $(x_d, y_d, z_d), (\phi_d, \theta_d, \psi_d)$ représentent respectivement la position et l'attitude désirées.

$$\begin{cases} e_x = x_d - x \\ e_y = y_d - y \\ e_z = z_d - z \\ e_\phi = \phi_d - \phi \\ e_\theta = \theta_d - \theta \\ e_\psi = \psi_d - \psi \end{cases} \quad (5-2)$$

Afin de limiter l'effort de la commande, l'action dérivée agira sur les états au lieu d'agir sur leurs erreurs telle qu'illustré à la Figure 5.2. Cette manière empêchera une discontinuité possible au niveau des trajectoires désirées de s'amplifier par l'action dérivée [31]. Certes, les positions désirées (x_d, y_d, z_d) et le lacet ψ_d seront des signaux continus pour simuler un vol stationnaire. Cependant, la forme des signaux (ϕ_d, θ_d) n'est pas fixée au préalable, ces signaux de référence sont générés par le contrôleur de la boucle externe.

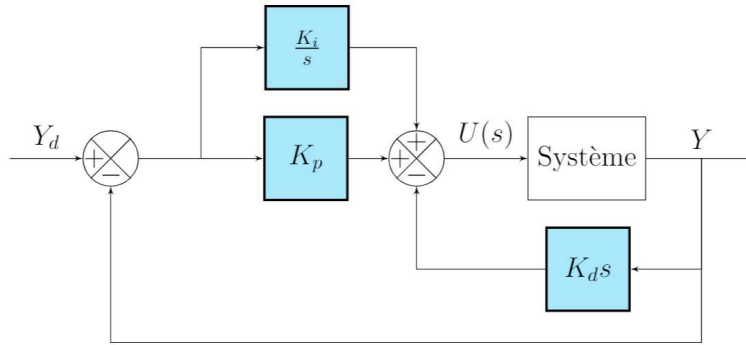


Figure 5.2: Contrôleur PID

La stabilisation de l'altitude z en une référence z_d est assurée par le signal de commande T , qui représente la force de poussée nécessaire pour atteindre z_d :

$$T = k_{pz}e_z - k_{dz}\dot{e}_z + k_{iz}\int e_z \quad (5-3)$$

Le mouvement latéral est généré par l'inclinaison du quadrotor autour de l'axe de roulis. La structure de la commande de ce mouvement est composée alors de deux boucles en cascade comme le montre la Figure 5.3. Le contrôleur de la boucle externe produit la référence de l'orientation du roulis ϕ_d , tandis que celui de la boucle interne assure le suivi de cette référence grâce au moment généré τ_ϕ . Ainsi, le drone est guidé vers la référence du mouvement latéral y_d . Dans ce cas, les lois de commande externe et interne s'écrivent:

$$\begin{cases} \phi_d = k_{py}e_y - k_{dy}\dot{y} \\ \tau_\phi = k_{p\phi}e_\phi - k_{d\phi}\dot{\phi} \end{cases} \quad (5-4)$$

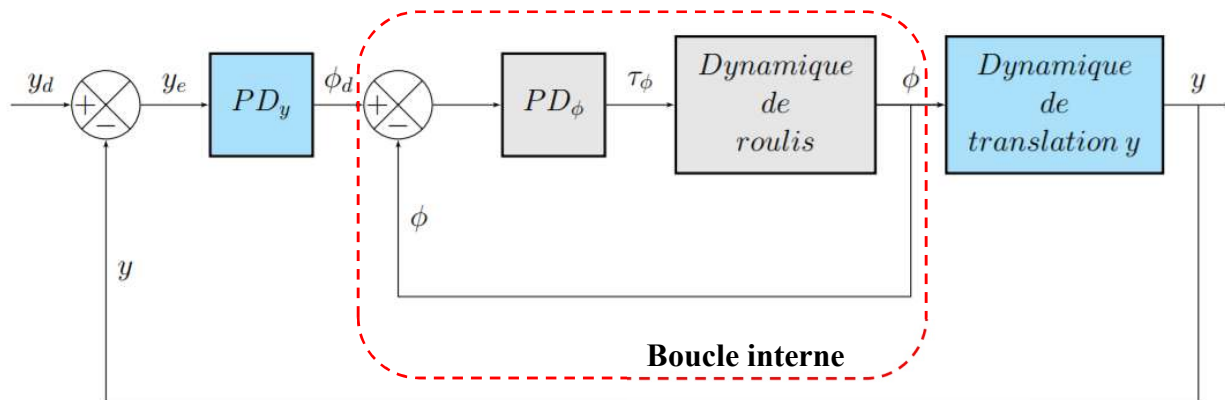


Figure 5.3: Le mouvement latéral du quadrotor

Le même principe s'applique au mouvement longitudinal comme le présente la Figure 5.4. Ce mouvement est généré par l'inclinaison du quadrotor autour de son axe de tangage. La boucle externe produit la référence de l'orientation θ_d , tandis que la boucle interne guide le drone à conserver cette orientation de tangage grâce au moment produit τ_θ . Il en résulte la poursuite de la trajectoire longitudinale de référence x_d . Les lois de commande externe et interne sont les suivantes:

$$\begin{cases} \theta_d = k_{px}e_x - k_{dx}\dot{x} \\ \tau_\theta = k_{p\theta}e_\theta - k_{d\theta}\dot{\theta} \end{cases} \quad (5-5)$$

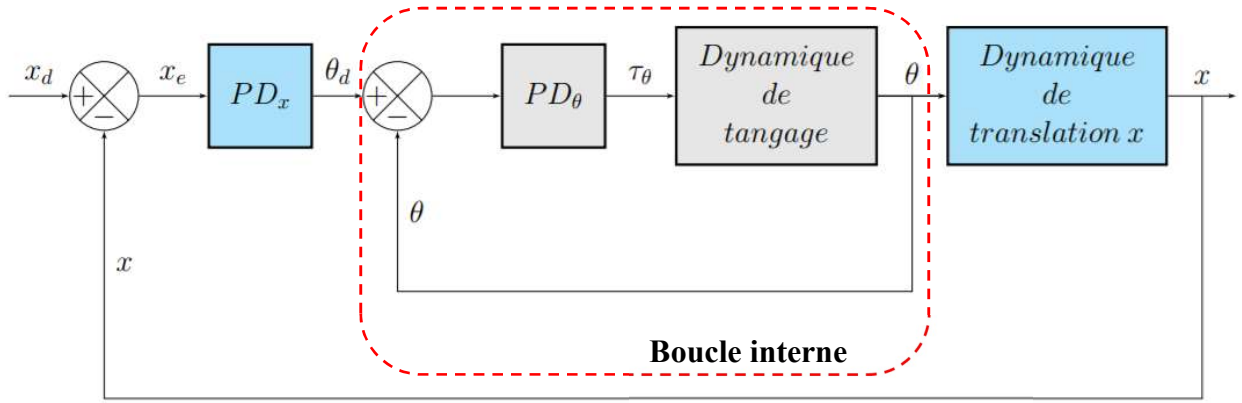


Figure 5.4: Le mouvement longitudinal du quadrotor

La dernière loi de contrôle concerne la stabilisation du mouvement de lacet ψ à une référence ψ_d en régime permanent grâce au moment τ_ψ :

$$\tau_\psi = k_{p\psi}e_\psi - k_{d\psi}\dot{\psi} \quad (5-6)$$

Il est intéressant de mentionner que la stabilisation du roulis et du tangage doit se faire plus rapidement que la dynamique longitudinale et latérale pour pouvoir stabiliser le quadrotor en une position désirée.

Les valeurs des gains de ce contrôleur seront données ultérieurement dans ce travail.

5.2 Contrôle du quadrotor par backstepping

Dans cette section, nous nous intéressons au contrôleur par backstepping, développé dans le mémoire de Guillaume Charland [25], pour des fins de comparaison.

Le modèle du quadrotor dans ce travail est analogue à celui sur lequel ce mémoire se base. Un contrôleur par backstepping y est construit en quatre étapes. De prime abord, l'étape de contrôle de la position linéaire puis le contrôle de la vitesse linéaire, ensuite le contrôle de la position angulaire et enfin le contrôle de la vitesse angulaire. Un estimateur de dérivée y est utilisé en chaque étape pour estimer les dérivées des commandes virtuelles (se référer à [25] pour plus de détails). Les commandes virtuelles ainsi que les sorties du contrôleur sont données comme suit.

Les signaux nécessaires aux calculs sont :

$$\begin{aligned}
 \eta_e &= \eta - \eta_d \\
 \mu_e &= \mu - \alpha_\mu \\
 \Theta_e &= \Theta - \alpha_\Theta \\
 \omega_e &= \omega - \alpha_\omega
 \end{aligned} \tag{5-7}$$

Les commandes virtuelles sont :

$$\begin{aligned}
 \alpha_\mu &= \dot{\eta}_d - \Lambda_1 \eta_e \\
 \alpha_a &= [\alpha_{a_x} \alpha_{a_y} \alpha_{a_z}]^T = \eta_e + g \cdot k - \ddot{\eta}_d + \Lambda_1(\mu_e - \Lambda_1 \eta_e) + \Lambda_2 \mu_e \\
 \alpha_\omega &= \rho(\Theta)^{-1}(\hat{\alpha}_\Theta - \Lambda_3 \Theta_e) \\
 \alpha_\phi &= \sin^{-1} \left(\frac{\alpha_{a_x} \sin(\alpha_\psi) - \alpha_{a_y} \cos(\alpha_\psi)}{\sqrt{\alpha_a^T \alpha_a}} \right) \\
 \alpha_\phi &= \tan^{-1} \left(\frac{\alpha_{a_x} \cos(\alpha_\psi) + \alpha_{a_y} \sin(\alpha_\psi)}{\alpha_{a_z}} \right) \\
 \alpha_\Theta &= [\alpha_\phi \ \alpha_\theta \ \psi_d]^T
 \end{aligned} \tag{5-8}$$

Les commandes réelles (c.-à-d. les sorties du contrôleur) sont :

$$\begin{aligned}
T &= m\sqrt{\alpha_a^T \alpha_a} \\
\tau &= -\rho(\Theta)^T \Theta_e + \omega \times I \omega + G_r + I \hat{\alpha}_\omega - \Lambda_4 \omega_e
\end{aligned} \tag{5-9}$$

L'estimateur de dérivée, où l'indice $i \in \{\phi, \theta, p, q, r\}$

$$\begin{aligned}
\hat{\alpha}_i &= \xi_i - \lambda_i \sqrt{|\alpha_i - \hat{\alpha}_i|} \text{sign}(\alpha_i - \hat{\alpha}_i) \\
\dot{\xi}_i &= -\beta_i \text{sign}(\alpha_i - \hat{\alpha}_i)
\end{aligned} \tag{5-10}$$

où $\Lambda_1, \Lambda_2, \Lambda_3, \Lambda_4, \lambda_i$ et β_i sont des paramètres strictement positifs.

5.3 Contrôle du quadrotor par backstepping avec action intégrale

Dans cette section, la procédure du contrôle par backstepping est modifiée afin d'introduire une action intégrale. Cette action permettra de venir à bout des erreurs résiduelles dues aux perturbations ou bien aux incertitudes. Grâce à cette stratégie, la démonstration de la stabilité globale du système commandé est possible. Outre les performances en régime permanent que procurera ce type de modification, le nouveau contrôleur doté de l'action intégrale donnera lieu à une forme linéaire du système en boucle fermée. Cette forme sera exploitée par la suite dans l'analyse de la robustesse aux incertitudes de la masse par le biais d'outils d'analyse linéaires.

Les étapes de conception dans cette section considèrent l'erreur ainsi que son intégrale à la différence du backstepping présenté en [25]. Chaque écart e_i est remplacé par l'expression qui suit :

$$e_i = (x_i - x_{i,r}) + K \int (x_i - x_{i,r}) \tag{5-11}$$

où K est un gain positif à régler, x_i est l'état considéré et $x_{i,r}$ est le signal de référence lui étant associé.

L'objectif présent du contrôle est de permettre au quadrotor la poursuite de la trajectoire désirée de la position $\boldsymbol{\eta}_d = [\mathbf{x}_d, \mathbf{y}_d, \mathbf{z}_d]$ et du lacet $\boldsymbol{\psi}_d$. À cet effet, nous nous basons sur le modèle du véhicule (4-14) qui décrit sa dynamique dans le repère inertiel. Ce modèle peut être décomposé en deux sous-systèmes interconnectés grâce aux angles d'Euler dont dépend la matrice de rotation R

(voir A.1 Annexe 1), le premier sous-système lié au mouvement de translation et le deuxième lié au mouvement de rotation telle que représentée par les équations (5-12).

$$\Sigma_T := \begin{cases} \dot{\eta} = \mu \\ \dot{\mu} = -\frac{1}{m}TR\mathbf{k} + g\mathbf{k} \end{cases} \quad (5-12)$$

$$\Sigma_R := \begin{cases} \dot{\Theta} = \rho(\Theta)\omega \\ I\dot{\omega} = \tau - G_r - [\omega]_{\times}I\omega \end{cases}$$

En vue de déterminer les commandes T et τ nécessaires à l'atteinte de notre objectif, nous commençons par le développement du contrôleur des positions x, y et z , et synthétisons par la suite le contrôleur de l'orientation décrite par les angles d'Euler ϕ, θ , et ψ .

5.3.1 Contrôle du système de positionnement

Nous considérons d'abord le système Σ_T pour ressortir la force de poussée totale T permettant de stabiliser le quadrotor à l'altitude désirée et les angles de référence assurant le déplacement désiré horizontal. Ceci est réalisé en deux étapes.

Étape 1: Contrôle de la position

Étant donné η_d la trajectoire que l'on désire suivre, définissons l'erreur de poursuite e_1 et son intégrale χ_1 .

$$e_1 = \eta_d - \eta \quad (5-13)$$

$$\chi_1 = \int (\eta_d - \eta) \quad (5-14)$$

Soit la fonction candidate de Lyapunov du système:

$$V_1 = \frac{1}{2}e_1^T e_1 + \frac{1}{2}\chi_1^T K_1 \chi_1 \quad (5-15)$$

où K_1 est une matrice définie positive qui sera fixée par la suite.

La dérivée de V_1 le long de la solution du système s'écrit :

$$\dot{V}_1 = e_1^T (\dot{\eta}_d - \ddot{\eta} + K_1^T \chi_1) \quad (5-16)$$

Selon le principe de fonctionnement du backstepping, nous allons devoir contrôler la position à travers la vitesse. Le choix de la commande virtuelle $\alpha_{\dot{\eta}}$, représentant la vitesse désirée, permet d'obtenir \dot{V}_1 définie négative. Un tel choix est donné par :

$$\alpha_{\dot{\eta}} = \dot{\eta}_d + K_1^T \chi_1 + A_1 e_1 \quad (5-17)$$

où A_1 est une matrice définie positive réglable.

Étape 2: Contrôle de la vitesse linéaire

Soit e_2 l'écart entre la vitesse linéaire du drone et la commande virtuelle $\alpha_{\dot{\eta}}$.

$$e_2 = \alpha_{\dot{\eta}} - \dot{\eta} \quad (5-18)$$

À partir des équations (5-17), nous obtenons la dynamique de l'erreur \dot{e}_2 .

$$\begin{aligned} \dot{e}_2 &= \dot{\alpha}_{\dot{\eta}} - \ddot{\eta} \\ &= \ddot{\eta}_d + K_1^T e_1 + A_1 (e_2 - A_1 e_1 - K_1^T \chi_1) - \ddot{\eta} \end{aligned} \quad (5-19)$$

En cette étape, l'erreur e_2 doit être stabilisée. Pour cela, nous choisissons la fonction candidate de Lyapunov suivante :

$$V_2 = V_1 + \frac{1}{2} e_2^T e_2 \quad (5-20)$$

$$\begin{aligned} \dot{V}_2 &= e_1^T \dot{e}_1 + \chi_1^T K_1 e_1 + e_2^T \dot{e}_2 \\ &= e_1^T (e_2 - A_1 e_1 - K_1^T \chi_1) + \chi_1^T K_1 e_1 + e_2^T (\ddot{\eta}_d + K_1^T e_1 \\ &\quad + A_1 (e_2 - A_1 e_1 - K_1^T \chi_1) - \ddot{\eta}) \\ &= -e_1^T A_1 e_1 + e_2^T (\ddot{\eta}_d + (I - A_1^2 + K_1^T) e_1 + A_1 e_2 - A_1 K_1^T \chi_1 - \ddot{\eta}) \end{aligned} \quad (5-21)$$

À l'aide du terme $\ddot{\eta}$, nous forcerons \dot{V}_2 à être définie négative. Soit $\alpha_{\ddot{\eta}}$ la valeur désirée pour $\ddot{\eta}$, cette valeur nous permettra de calculer la poussée totale T et les angles de roulis et de tangage désirés.

$$\alpha_{\ddot{\eta}} = \ddot{\eta}_d + (I - A_1^2 + K_1^T)e_1 + (A_1 + A_2)e_2 - A_1 K_1^T \chi_1 \quad (5-22)$$

Rappelons que $\alpha_{\ddot{\eta}}$ constitue la commande virtuelle de l'accélération, elle représente réellement l'accélération du quadrotor dans le repère inertiel permettant le suivi de la trajectoire désirée η_d . Les composantes de ce vecteur sont écrites en (5-23).

$$\begin{aligned} \alpha_{\ddot{\eta}} &= g\mathbf{k} - \frac{T}{m}R\mathbf{k} \\ &= \begin{bmatrix} -\frac{1}{m}T(\cos(\phi_d)\sin(\theta_d)\cos(\psi_d) + \sin(\phi_d)\sin(\psi_d)) \\ -\frac{1}{m}T(\cos(\phi_d)\sin(\theta_d)\sin(\psi_d) - \sin(\phi_d)\cos(\psi_d)) \\ g - \frac{1}{m}T\cos(\phi_d)\cos(\theta_d) \end{bmatrix} \end{aligned} \quad (5-23)$$

En tenant compte de (5-23), il est perceptible que l'orientation du quadrotor détermine la direction de la force de poussée dans l'espace. Dès lors, il est primordial d'avoir une orientation correcte du quadrotor pour pouvoir atteindre la position désirée, et par conséquent, il est nécessaire de concevoir un contrôleur d'attitude à haute performance.

Nous allons adopter la notation $\alpha_{\ddot{\eta}} = \begin{bmatrix} \frac{1}{m}\alpha_{\ddot{x}} \\ \frac{1}{m}\alpha_{\ddot{y}} \\ g - \frac{1}{m}\alpha_{\ddot{z}} \end{bmatrix}$. Les angles ϕ_d et θ_d se calculent via les

équations:

$$\begin{aligned} \phi_d &= \sin^{-1} \left(-\frac{\alpha_{\ddot{x}} \sin(\psi_d) - \alpha_{\ddot{y}} \cos(\psi_d)}{\sqrt{(\alpha_{\ddot{x}}^2 + \alpha_{\ddot{y}}^2 + \alpha_{\ddot{z}}^2)}} \right) \\ \theta_d &= \tan^{-1} \left(\frac{\alpha_{\ddot{x}} \cos(\psi_d) + \alpha_{\ddot{y}} \sin(\psi_d)}{\alpha_{\ddot{z}}} \right) \end{aligned} \quad (5-24)$$

La poussée totale T se calcule selon :

$$T = \sqrt{\alpha_x^2 + \alpha_y^2 + \alpha_z^2} \quad (5-25)$$

Reprenons l'équation en (5-22), celle-ci peut être écrite différemment en ne considérant que la variable d'écart e_1 :

$$\begin{aligned} \alpha_{\ddot{\eta}} &= \ddot{\eta}_d + (I - A_1^2 + K_1^T)e_1 + (A_1 + A_2)e_2 - A_1 K_1^T \chi_1 \\ &= \ddot{\eta}_d + A_2 K_1^T \int e_1 + (I + K_1^T + A_2 A_1)e_1 + (A_1 + A_2)\dot{e}_1 \end{aligned} \quad (5-26)$$

On pose $\Gamma_1 = A_2 K_1^T, \Gamma_2 = I + K_1^T + A_2 A_1, \Gamma_3 = A_1 + A_2$, l'expression de l'accélération désirée devient :

$$\alpha_{\ddot{\eta}} = \ddot{\eta}_d + \Gamma_1 \int e_1 + \Gamma_2 e_1 + \Gamma_3 \dot{e}_1 \quad (5-27)$$

Nous constatons que le contrôleur de position se présente sous une structure d'un contrôleur PID parallèle.

5.3.2 Contrôle du système d'orientation

Le contrôleur à concevoir dans cette partie doit être capable de suivre l'orientation désirée $\Theta_d = [\phi_d, \theta_d, \psi_d]^T$, dont les deux premiers éléments résultent du calcul effectué dans la section précédente et le troisième élément est une trajectoire désignée par l'utilisateur. À la fin de cette sous-section, nous aurons calculé le moment $\tau = [\tau_\phi, \tau_\theta, \tau_\psi]^T$.

Étape 1: Contrôle de la position angulaire

Similairement au contrôle de la position, nous commençons par la définition de l'erreur d'orientation e_3 ainsi que son intégrale.

$$e_3 = \Theta_d - \Theta \quad (5-28)$$

$$\chi_2 = \int (\Theta_d - \Theta) \quad (5-29)$$

Avec le choix de la fonction candidate de Lyapunov V_3 .

$$V_3 = \frac{1}{2} e_3^T e_3 + \frac{1}{2} \chi_2^T K_2 \chi_2 \quad (5-30)$$

où K_2 est une matrice définie positive. La dérivée de V_3 par rapport au temps est :

$$\begin{aligned}\dot{V}_3 &= e_3^T \dot{e}_3 + \chi_2^T K_2 e_3 \\ \dot{V}_3 &= e_3^T (\dot{\Theta}_d - \dot{\Theta} + K_2^T \chi_2) \\ &= e_3^T (\dot{\Theta}_d - \rho \omega + K_2^T \chi_2)\end{aligned}\tag{5-31}$$

Nous utilisons la vitesse angulaire ω pour contrôler la position angulaire. Un choix judicieux d'une vitesse angulaire désirée α_ω (commande virtuelle) rendra négatif la dérivée de V_3 . Un tel choix, où A_3 est une matrice définie positive, est donné comme suit :

$$\alpha_\omega = \rho^{-1}(\dot{\Theta}_d + K_2^T \chi_2 + A_3 e_3)\tag{5-32}$$

Lorsque ω converge vers α_ω , la dérivée de la candidate de Lyapunov devient :

$$\dot{V}_3 = -e_3^T A_3 e_3\tag{5-33}$$

Il en découle que le système en (5-28) est asymptotiquement stable.

Étape 2: Contrôle de la vitesse angulaire

La vitesse angulaire doit être contrôlée en cette étape. Pour ce faire, nous disposons directement de la commande réelle τ qui représente le vecteur moment.

Définissons la variable e_4 qui représente l'écart entre la vitesse angulaire du quadrotor et la commande virtuelle α_ω .

$$e_4 = \alpha_\omega - \omega\tag{5-34}$$

Soit la fonction candidate de Lyapunov

$$V_4 = V_3 + \frac{1}{2} e_4^T I e_4\tag{5-35}$$

où I est la matrice d'inertie du quadrotor.

Afin de calculer la dérivée de V_4 par rapport au temps, nous aurons besoin de la dynamique de l'erreur e_4 . En utilisant l'expression de $\dot{\omega}$ (4-14), \dot{e}_4 est donnée par :

$$\dot{e}_4 = \dot{\alpha}_\omega - \dot{\omega} = \dot{\alpha}_\omega - I^{-1}(\tau - \omega \times I\omega - G_r) \quad (5-36)$$

La matrice I dans l'équation précédente représente la matrice d'inertie de l'aéronef, qui est une matrice définie positive toujours inversible.

En utilisant les équations (5-35) et (5-32), la dérivée de V_4 devient :

$$\begin{aligned} \dot{V}_4 &= e_3^T \dot{e}_3 + \chi_2^T K_2 e_3 + e_4^T I \dot{e}_4 \\ &= e_3^T (\dot{\Theta}_d - \rho\omega + K_2^T \chi_2) + e_4^T I (\dot{\alpha}_\omega - I^{-1}(\tau - \omega \times I\omega - G_r)) \\ &= -e_3^T A_3 e_3 + e_4^T I (\dot{\alpha}_\omega - I^{-1}(\tau - \omega \times I\omega - G_r) + I^{-1} \rho^T e_3) \\ &= -e_3^T A_3 e_3 + e_4^T (I \dot{\alpha}_\omega - (\tau - \omega \times I\omega - G_r) + \rho^T e_3) \end{aligned} \quad (5-37)$$

Le choix de τ est effectué pour rendre \dot{V}_4 négative, en choisissant

$$\tau = (I \dot{\alpha}_\omega + \omega \times I\omega + G_r + \rho^T e_3 + A_4 e_4) \quad (5-38)$$

où A_4 est une matrice de conception définie positive. Ce qui implique :

$$\dot{V}_4 = -e_3^T A_3 e_3 - e_4^T A_4 e_4 \quad (5-39)$$

La stabilité asymptotique du sous-système en boucle fermée est alors assurée.

5.3.3 Le système en boucle fermée

La fonction candidate de Lyapunov du système complet est :

$$V = \frac{1}{2} e_1^T e_1 + \frac{1}{2} \chi_1^T K_1 \chi_1 + \frac{1}{2} e_2^T e_2 + \frac{1}{2} e_3^T e_3 + \frac{1}{2} \chi_2^T K_2 \chi_2 + \frac{1}{2} e_4^T e_4 \quad (5-40)$$

Sa dérivée par rapport au temps est :

$$\dot{V} = -e_1^T A_1 e_1 - e_2^T A_2 e_2 - e_3^T A_3 e_3 - e_4^T A_4 e_4 \quad (5-41)$$

Il en découle que le système en boucle fermée est asymptotiquement stable au sens de Lyapunov.

D'autre part, nous allons essayer d'écrire le système Σ_T en boucle fermée sous forme matricielle pour des raisons qui deviendront claires dans des sections ultérieures.

À partir de (5-26), le système Σ_T en boucle fermée est représenté par :

$$\begin{bmatrix} e_1 \\ \dot{e}_1 \\ \ddot{e}_1 \end{bmatrix} = \begin{bmatrix} 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & I_3 \\ -\Gamma_1 & -\Gamma_2 & -\Gamma_3 \end{bmatrix} \begin{bmatrix} \int e_1 \\ e_1 \\ \dot{e}_1 \end{bmatrix} \quad (5-42)$$

où $\Gamma_1 := A_2 K_1^T$, $\Gamma_2 := I + K_1^T + A_2 A_1$ et $\Gamma_3 := A_1 + A_2$. Les paramètres A_1 , A_2 et K_1 sont à déterminer selon la position désirée des pôles en boucle fermée.

5.3.4 Signaux de commande

Nous résumons ici tous les signaux nécessaires au contrôle dans cette section.

Les signaux d'erreur sont :

$$e_1 = \eta_d - \eta ; \chi_1 = \int (\eta_d - \eta) \quad (5-43)$$

$$e_2 = \alpha_{\dot{\eta}} - \dot{\eta} \quad (5-44)$$

$$e_3 = \Theta_d - \Theta ; \chi_2 = \int (\Theta_d - \Theta) \quad (5-45)$$

$$e_4 = \alpha_{\omega} - \omega \quad (5-46)$$

Les commandes virtuelles sont :

$$\alpha_{\dot{\eta}} = \dot{\eta}_d + K_1^T \chi_1 + A_1 e_1 \quad (5-47)$$

$$\alpha_{\omega} = \rho^{-1}(\dot{\Theta}_d + K_2^T \chi_2 + A_3 e_3) \quad (5-48)$$

$$\alpha_{\ddot{\eta}} = \begin{bmatrix} \frac{1}{m} \alpha_{\ddot{x}} \\ \frac{1}{m} \alpha_{\ddot{y}} \\ g - \frac{1}{m} \alpha_{\ddot{z}} \end{bmatrix} = \ddot{\eta}_d + (I - A_1^2 + K_1^T) e_1 + (A_1 + A_2) e_2 - A_1 K_1^T \chi_1 \quad (5-49)$$

Les commandes réelles sont :

$$\phi_d = \sin^{-1} \left(-\frac{\alpha_{\ddot{x}} \sin(\psi_d) - \alpha_{\ddot{y}} \cos(\psi_d)}{\sqrt{(\alpha_{\ddot{x}}^2 + \alpha_{\ddot{y}}^2 + \alpha_{\ddot{z}}^2)}} \right), \theta_d = \tan^{-1} \left(\frac{\alpha_{\ddot{x}} \cos(\psi_d) + \alpha_{\ddot{y}} \sin(\psi_d)}{\alpha_{\ddot{z}}} \right) \quad (5-50)$$

$$\tau = (I\dot{\alpha}_\omega + \omega \times I\omega + G_r + \rho^T e_3 + A_4 e_4) \quad (5-51)$$

$$T = \sqrt{\alpha_{\ddot{x}}^2 + \alpha_{\ddot{y}}^2 + \alpha_{\ddot{z}}^2} \quad (5-52)$$

Nous avons détaillé dans ce chapitre la conception des contrôleurs dont sera muni le quadrotor. Dans un premier lieu, un contrôleur PID a été développé en se basant sur le modèle linéaire du quadrotor pendant le vol stationnaire. En deuxième lieu, le backstepping avec action intégrale a été appliqué au cas du quadrotor. Les performances de ces contrôleurs seront évaluées par simulation dans le sixième chapitre. Le choix des gains Γ_1, Γ_2 et Γ_3 du contrôleur non linéaire sera effectué dans le chapitre suivant en utilisant les applications gardiennes.

CHAPITRE 6 APPLICATIONS GARDIENNES

6.1 Problématique de robustesse

Nous avons conçu jusqu'à présent un contrôleur qui rend le système en boucle fermée asymptotiquement stable, en supposant que le système est parfaitement connu.

La résolution de la problématique industrielle que nous abordons dans ce projet exige que le contrôleur développé soit performant également sur des quadrotors de masses variables dans un intervalle $[m_1, m_2]$, où les valeurs de m_1 et m_2 sont à préciser.

La dynamique du quadrotor révèle qu'une potentielle variation de la masse influera la dynamique de translation du quadrotor. Quant à la dynamique de l'orientation, qui dépend de la matrice d'inertie du véhicule, elle est peu affectée par un tel changement [32]. D'autre part, la compagnie présentant l'exigence technique dispose actuellement d'un contrôleur bas niveau (contrôleur de l'orientation) qui a prouvé son efficacité, l'intérêt majeur penche vers le contrôleur haut niveau (contrôleur de la position).

Nous nous focalisons désormais sur le sous-système décrivant la dynamique de translation (Σ_T). À cette étape, nous admettons que le contrôleur du mouvement de rotation est à haute performance. Nous nous intéressons à déduire les gains du contrôleur du mouvement de translation capables de garantir à la dynamique de l'erreur de position des performances souhaitées en cas de variation de la masse du quadrotor dans un intervalle donné. Pour cet effet, la masse de l'engin sera considérée comme un paramètre incertain dans la représentation du quadrotor en boucle fermée. Ceci permettra une analyse de robustesse du contrôleur concerné à l'aide des applications gardiennes.

6.2 Présentation théorique

Cette partie nous sert d'introduction de notions élémentaires sur les applications gardiennes, ces dernières, développées par Saydy en 1990, constituent un outil pour l'étude de la stabilité générale de familles matricielles ou polynomiales. Les définitions et les propositions présentées dans cette section proviennent essentiellement de [33] (voir aussi [34] et [35]). Il est important de noter que nous nous intéressons dans ce qui suit aux applications gardiennes appliquées à des matrices. Toutefois les mêmes définitions et propositions s'appliquent à l'étude des polynômes.

6.2.1 Définitions et propriétés

Les applications gardiennes permettent la détermination des ensembles des paramètres pour lesquels la stabilité généralisée d'un système est garantie.

Définition 6.1. Soit Ω un sous-ensemble ouvert du plan complexe \mathbb{C} et $\partial\Omega$ sa frontière.

Une matrice A est dite Ω -stable si ses valeurs propres se trouvent à l'intérieur du domaine Ω , c.-à-d. $\sigma(A) = \{\lambda_1, \lambda_2, \dots, \lambda_n\} \subset \Omega$, où les λ_i sont des valeurs propres de A . Dans ce cas, on écrit $A \in S(\Omega) := \{A \in \mathbb{R}^{n \times n} | \sigma(A) \subset \Omega\}$.

Définition 6.2. Soit X l'ensemble des matrices carrées réelles de taille n et soit S un sous-ensemble ouvert de X . Soit v une application de X dans \mathbb{C} . On dit que v garde S , si pour tout $A \in \bar{S}$, on a :

$$v(A) = 0 \Leftrightarrow A \in \partial S \quad (6-1)$$

où \bar{S} représente l'adhérence de S . On dit aussi que v est une application gardienne pour S , lequel est un ensemble de stabilité généralisée du type $S(\Omega)$ pour un domaine Ω donné.

Proposition 6.1 Soit une famille de matrices $A(r)$, paramétrée de façon continue, $r = (r_1, r_2, \dots, r_m) \in U$, avec U un ensemble connexe.

Soit l'ensemble S gardé par l'application gardienne v et supposons que $A(r_0)$ est nominale Ω -stable c.-à-d. $A(r_0) \in S$ pour un $r_0 \in U$. Alors :

$$A(r) \in S \quad \forall r \in U \Leftrightarrow v(A(r)) \neq 0 \quad \forall r \in U \quad (6-2)$$

Cela dit qu'une application est gardienne d'un domaine Ω si et seulement si elle s'annule uniquement lorsqu'une des valeurs propres est à la frontière de Ω .

Cette proposition nous permet la conclusion du confinement des pôles de $A(r)$ dans S lorsque r parcourt U . Le corollaire qui s'en suit complète cette proposition, il considère cette fois l'ensemble U des paramètres de la famille.

Corollaire 6.1 Soit $S(\Omega)$ gardé par l'application v et la famille $\{A(r) : r \in U\}$. Alors l'ensemble C défini par $C = \{r \in U : v(A(r)) = 0\}$ divise l'espace des paramètres en composantes C_i qui sont

stables ou instables par rapport à Ω . Il suffit de tester un vecteur r_i de chacune des composantes C_i pour savoir si celle-ci est stable ou pas.

Ce résultat est considérable. L'espace des paramètres sera alors découpé en régions, et la vérification de la stabilité d'un seul point de l'intérieur de chaque région nous permettra de conclure de sa stabilité ou non.

Nous considérons par la suite une famille de matrices polynomiales $A(r)$ dépendant d'un seul paramètre r (la masse m dans le cas de notre étude). On suppose qu'il existe un r_0 tel que $A(r_0)$ soit stable par rapport à Ω .

Nous cherchons le plus grand intervalle ouvert contenant r_0 tel que $A(r)$ soit stable par rapport à Ω . Soit :

$$\begin{aligned} r_{min} &= \max \{r < r_0 : v_{\Omega}(A(r)) = 0\} \text{ } (-\infty \text{ sinon}) \\ r_{max} &= \min \{r > r_0 : v_{\Omega}(A(r)) = 0\} \text{ } (+\infty \text{ sinon}) \end{aligned} \tag{6-3}$$

Lemme 6.1 Soit $A(r)$ une matrice polynomiale dépendant du paramètre incertain r telle que $A(r_0)$ soit Ω -stable. Soit $S(\Omega)$ gardé par l'application v_{Ω} . Alors $A(r)$ est Ω -stable pour tout $r \in]r_{min}, r_{max}[$.

Par ailleurs, cet intervalle est l'intervalle maximal de stabilité contenant r_0 .

En particulier, r_{min} (respectivement r_{max}) est la première valeur de r à gauche (respectivement à droite) de r_0 pour laquelle le système se déstabilise; et l'on peut conclure que, forcément, $A(r_{min})$ (respectivement $A(r_{max})$) est instable avec des valeurs sur $\delta\Omega$ et aucune en dehors de Ω .

6.2.2 Applications gardiennes usuelles

Il existe une procédure de calcul des applications gardiennes dont l'ensemble à garder possède une frontière qui se décrit par un polynôme x et y du plan complexe, nous proposons pour cela la consultation de [33] pour plus de détails. Dans cette partie, nous allons parcourir certains domaines classiques de stabilité, illustrés à la Figure 6.1, en présentant pour chacun d'eux une application gardienne.

Tout d'abord, nous définissons le produit bialterné de deux matrices qui nous servira à la construction des applications gardiennes.

Définition 6.3 Produit bialterné.

Le produit bialterné est défini pour deux matrices carrées A et B de taille n par une matrice carrée $A \odot B$ de taille $\frac{n(n-1)}{2}$. Soit V^n le $\frac{n(n-1)}{2}$ -tuplet des paires d'entiers (p, q) , $p = 2, \dots, n$, $q = 1, \dots, p-1$, listé lexicalement, c'est-à-dire :

$$V^n = [(2,1), (3,1), (3,2), (4,1), (4,2), (4,3), (5,1), \dots, (n, n-1)]$$

Soit V_i^n la $i^{\text{ème}}$ composante de V^n et soit

$$f((p, q), (r, s)) := \frac{1}{2} \left(\det \begin{bmatrix} a_{pr} & a_{ps} \\ b_{qr} & b_{qs} \end{bmatrix} + \det \begin{bmatrix} b_{pr} & b_{ps} \\ a_{qr} & a_{qs} \end{bmatrix} \right) \quad (6-4)$$

Les éléments du produit bialterné de A et B sont donnés par :

$$(A \odot B)_{ij} = f(V_i^n, V_j^n) \quad (6-5)$$

À partir de ce produit bialterné, il est possible de construire les applications gardiennes énoncées ci-dessous :

- **Stabilité d'Hurwitz:** Elle correspond au placement des pôles dans le demi-plan gauche ouvert. Le domaine qui lui est accordé est $\Omega = \{z \in \mathbb{C} \mid \text{Re}(z) < 0\}$, une application gardienne possible pour $S(\Omega)$ est :

$$v_H(A) = \det(A \odot I) \det(A) \quad (6-6)$$

- **Marge de négativité:** elle correspond au demi-plan gauche ouvert décalé de α , le domaine Ω qui lui convient est $\Omega = \{z \in \mathbb{C} \mid \text{Re}(z) < \alpha < 0\}$, ce domaine peut être gardé par l'application suivante :

$$v_\alpha = \det(A \odot I - \alpha I \odot I) \det(A - \alpha I) \quad (6-7)$$

- **Cône d'amortissement:** ce cône est déterminé par l'angle limite d'écartement θ_0 , de telle manière que $\xi = \cos(\theta_0)$ désigne l'amortissement limite. Le domaine Ω correspondant est $\Omega = \{z \in \mathbb{C} \mid |\arg(z)| < \theta_0\}$. Une application gardienne est donnée par :

$$\nu_{\xi}(A) = \det(A^2 \odot I + (1 - 2\xi^2)A \odot A) \det(A) \quad (6-8)$$

- **Disque:** Ce disque limite la pulsation des pôles, le domaine $\Omega = \{z \in \mathbb{C} \mid |z| < \omega\}$ formé par le disque de rayon ω , est gardé par :

$$\nu_{\omega}(A) = \det(A \odot A - \omega^2 I \odot I) \det(A - \omega I) \det(A + \omega I) \quad (6-9)$$

Les applications que nous venons d'énumérer gardent un seul domaine à la fois. Habituellement, les contraintes à satisfaire lors de la synthèse d'une loi de commande concernent une combinaison des conditions usuelles, pour cette raison il est important d'avoir une méthode permettant l'obtention d'une application gardienne d'un domaine résultant de l'intersection des domaines usuels.

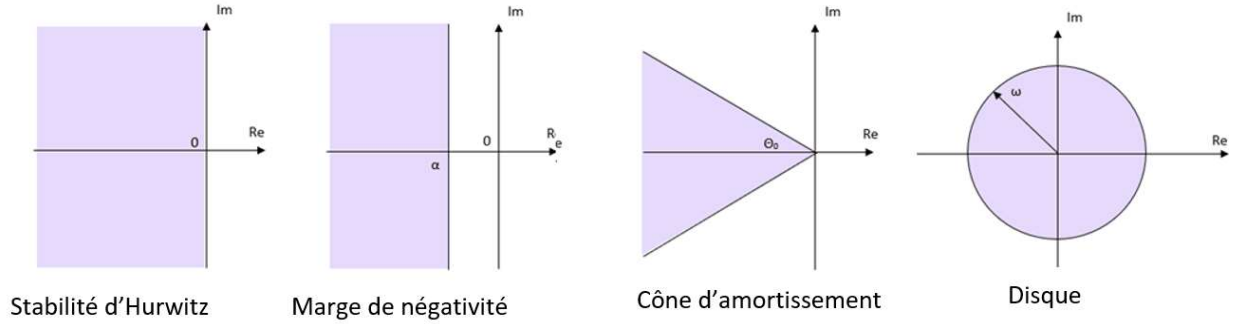


Figure 6.1: Domaines de stabilité usuels

La proposition suivante tirée de [33] permet d'atteindre cet objectif.

Proposition 6.2 Soit $\Omega = \Omega_1 \cap \Omega_2$. Si $S(\Omega_1)$ est gardé par ν_1 et $S(\Omega_2)$ par ν_2 .

Alors, $S(\Omega)$ est gardé par $\nu = \nu_1 \nu_2$.

6.2.3 Exemple d'application

Soit la famille de matrice bi paramétrique

$$\mathcal{A} = \{A(r_1, r_2) \mid (r_1, r_2) \in [-5, 5] \times [-3, 3]\}$$

La famille \mathcal{A} est donnée par :

$$A(r_1, r_2) = \begin{bmatrix} 0 & 1 \\ -4 + r_1 + r_2 & -3 + 4r_2 \end{bmatrix}$$

Nous souhaitons étudier le confinement des pôles de la famille \mathcal{A} à l'intérieur du cône d'amortissement $\xi = 0.7$ et à gauche de l'axe des imaginaires avec une marge de négativité $\alpha = -1$.

La matrice $A(0,0)$ a comme valeurs propres $\lambda_{1,2} = -1.5 \pm 1.32i$, par conséquent, la stabilité nominale est assurée pour $A(r_0)$ avec $r_0 = (0,0)$.

Les applications gardiennes correspondantes aux deux domaines d'intérêt obtenues symboliquement à l'aide de (6-7) et (6-8) sont données par:

$$\bullet v_\xi(A) = \det(A^2 \odot I + (1 - 2\xi^2)A \odot A) \det(A) = -(r_1 + r_2 - 4) \left(8r_2^2 - \frac{551}{50}r_2 + \frac{49}{50}r_1 + \frac{29}{50} \right) \quad (6-10)$$

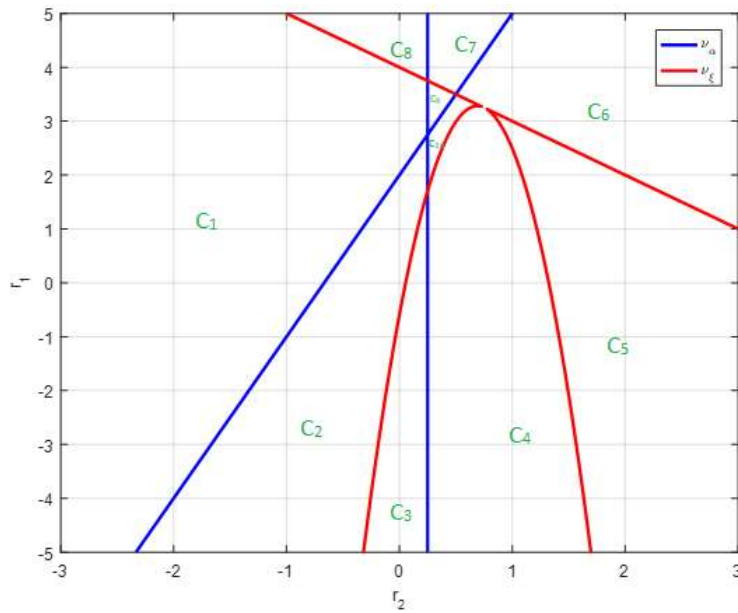
$$\bullet v_\alpha(A) = \det(A \odot I - \alpha I \odot I) \det(A - \alpha I) = \left(2r_2 - \frac{1}{2} \right) (3r_2 - r_1 + 2) \quad (6-11)$$

Nous traçons les contours de v_ξ et v_α ce qui permettra de décomposer le plan en plusieurs composantes délimitées par les points d'annulations de ces applications gardiennes, le résultat est représenté à la Figure 6.2.

Nous pouvons ainsi conclure que la région maximale contenant le point nominal r_0 dans l'espace des paramètres est l'ensemble C_2 délimité par $v(r_1, r_2) = 0$.

Si on s'intéresse à d'autres points nominaux, un point (r_1, r_2) est choisi à l'intérieur de chaque composante C_i du plan et les valeurs propres de $A(r_1, r_2)$ sont calculées. En se basant sur le Corollaire 6.1 la position des pôles trouvés nous dévoilera la stabilité ou l'instabilité de l'ensemble des points de la composante concernée. Les résultats obtenus sont rassemblés dans le Tableau 6.1.

À la lumière de ces résultats, nous pouvons déduire que seule la composante C_2 contient un point qui satisfait simultanément les conditions sur l'amortissement et la marge de négativité. Dès lors, nous concluons que la composante C_2 est stable puisque nominale Ω -stable.

Figure 6.2: Points d'annulation de ν_α et ν_ξ Tableau 6.1: Valeurs propres de la matrice A aux points (r_1, r_2)

Composante	(r_1, r_2)	Valeurs propres de $A(r_1, r_2)$	Ω -stabilité
C₁	(1,-2)	-10.52 et -0.48	Non
C₂	(-2,-1)	-5.79 et -1.2	Oui
C₃	(-2,0)	$-1.5 \pm 1.93 i$	Non
C₄	(-2,1)	$-0.5 \pm 2.18 i$	Non
C₅	(-1,2)	0.7 et 4.3	Non
C₆	(3,2)	-0.19 et 5.19	Non
C₇	(4, 0.35)	-1.8 et 0.19	Non
C₈	(4.5, 0)	-3.16 et 0.16	Non
C₉	(3.5, 0.33)	-1.57 et -0.1	Non
C₁₀	(2.8, 0.3)	$-0.9 \pm 0.3 i$	Non

Tel que mentionné plus haut, on s'intéresse dans ce travail à des familles monoparamétrique (notre paramètre d'intérêt étant la masse).

Par exemple, si l'on prend le même exemple que ci-dessus mais en fixant r_2 à sa valeur nominale ($r_2 = 0$). L'application gardienne obtenue divise l'espace des paramètres (l'axe r_1) en 3 composantes puisque $v(r_1) = v_\xi(r_1) v_\alpha(r_1)$ s'annule en $z_1 = -0.59$, $z_2 = 2$ et $z_3 = 4$ de sorte que l'intervalle maximal de Ω -stabilité est donnée par $]z_1, z_2[$.

6.3 Application au cas du quadrotor

Dans cette section, la synthèse d'une loi de commande respectant le cahier des charges sera présentée. La loi de commande basée sur le backstepping avec action intégrale a été réalisée au chapitre précédent, le choix des gains du contrôleur par méthode graphique sera effectué à présent.

Les applications gardiennes d'intérêt et la stratégie de synthèse seront détaillées.

6.3.1 Modèle incertain

Avant de pousser plus loin dans l'étude, il faut procéder au préalable à l'élaboration du modèle en fonction du paramètre incertain: la masse.

Soit m_N la masse nominale du quadrotor pour laquelle le contrôleur par backstepping avec action intégrale a été développé, et m sa masse réelle. Tenant compte de ces modifications, nous reprenons les équations (5-12) et (5-27). La dynamique de l'erreur de la position en boucle fermée, en présence d'incertitudes paramétriques, se met alors sous la forme matricielle suivante:

$$\begin{bmatrix} e_1 \\ \dot{e}_1 \\ \ddot{e}_1 \end{bmatrix} = \begin{bmatrix} 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & I_3 \\ -\frac{m_N}{m} \Gamma_1 & -\frac{m_N}{m} \Gamma_2 & -\frac{m_N}{m} \Gamma_3 \end{bmatrix} \begin{bmatrix} \int e_1 \\ e_1 \\ \dot{e}_1 \end{bmatrix} + \begin{bmatrix} 0_3 \\ 0_3 \\ \left(1 - \frac{m_N}{m}\right) I_3 \end{bmatrix} (\ddot{\eta}_d - g \cdot \mathbf{k}) \quad (6-12)$$

Les gains Γ_1 , Γ_2 et Γ_3 sont définis à la section 5.3.1, g représente la constante gravitationnelle, \mathbf{k} le vecteur $[0 \ 0 \ 1]^T$ et $\ddot{\eta}_d$ l'accélération de la trajectoire à poursuivre.

En posant $X_{BF} = [\int e_1, e_1, \dot{e}_1]^T$ et $r := \frac{m_N}{m}$ dont la valeur nominale est par conséquent égale à 1, le modèle en (6-12) devient:

$$\begin{cases} \dot{X}_{BF} = A_{BF}(r, \Gamma_1, \Gamma_2, \Gamma_3)X + B_{BF}(r)(\ddot{\eta}_d - g \cdot \mathbf{k}) \\ Y = X_{BF} \end{cases} \quad (6-13)$$

où

$$A_{BF}(r, \Gamma_1, \Gamma_2, \Gamma_3) = \begin{bmatrix} 0 & I_3 & 0 \\ 0 & 0 & I_3 \\ -r\Gamma_1 & -r\Gamma_2 & -r\Gamma_3 \end{bmatrix} \text{ et } B_{BF} = \begin{bmatrix} 0_3 \\ 0_3 \\ (1-r)I_3 \end{bmatrix} \quad (6-14)$$

La relation entre les matrices Γ_1, Γ_2 et Γ_3 et les matrices de conception A_1, A_2 et K_1 se présente comme suit:

$$\begin{aligned} \Gamma_1 &= A_2 K_1^T \\ \Gamma_2 &= I + K_1^T + A_2 A_1 \\ \Gamma_3 &= A_1 + A_2 \end{aligned} \quad (6-15)$$

Les matrices A_1, A_2 et K_1 sont des matrices définies positives. Dans la littérature, ces matrices sont souvent considérées diagonales dont les éléments de la diagonale sont égaux. Il en découle que l'on peut choisir les matrices Γ_1, Γ_2 et Γ_3 comme suit :

$$\Gamma_1 = \begin{bmatrix} \gamma_1 & 0 & 0 \\ 0 & \gamma_1 & 0 \\ 0 & 0 & \gamma_1 \end{bmatrix}, \Gamma_2 = \begin{bmatrix} \gamma_2 & 0 & 0 \\ 0 & \gamma_2 & 0 \\ 0 & 0 & \gamma_2 \end{bmatrix}, \Gamma_3 = \begin{bmatrix} \gamma_3 & 0 & 0 \\ 0 & \gamma_3 & 0 \\ 0 & 0 & \gamma_3 \end{bmatrix} \quad (6-16)$$

où $\gamma_i > 0$ pour $i = 1, 2, 3$

Nous nous intéressons par la suite à la détermination du triplet des gains $(\Gamma_1, \Gamma_2, \Gamma_3)$, i.e. $(\gamma_1, \gamma_2, \gamma_3)$, permettant au contrôleur le respect des performances souhaitées pour tout quadrotor de masse m tel que $m \in [m_1, m_2]$.

6.3.2 Stratégie d'analyse

La matrice A_{BF} dans (6-13) décrit la dynamique de l'erreur de la position η en boucle fermée. L'emplacement des valeurs propres de A_{BF} détermine la vitesse de convergence de l'erreur vers 0, la convergence étant d'autant plus rapide que ces valeurs sont grandes. La solution idéale serait alors de choisir des gains du contrôleur correspondant à des pôles très rapides, cette action causera toutefois la saturation des moteurs du quadrotor. Par conséquent, un compromis entre la rapidité de convergence et les limitations physiques du système se doit d'être respecté.

Pour ce faire, nous souhaitons confiner les pôles de la dynamique de l'erreur en présence d'une incertitude de la masse dans un domaine Ω défini par une marge de négativité (α_1, α_2) et un cône d'amortissement $\xi = \cos(\theta)$ (Figure 6.3).

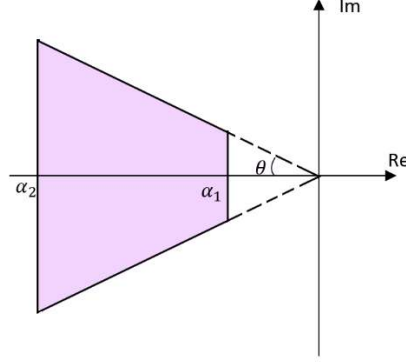


Figure 6.3: Le domaine Ω de stabilité généralisée

Il est à mentionner que le système en (6-15) présente une erreur en régime permanent non nulle due à l'incertitude, son expression est la suivante:

$$E_{RP} = \begin{bmatrix} \left(\frac{1}{r} - 1\right) \frac{\ddot{x}_d}{\gamma_1} \\ \left(\frac{1}{r} - 1\right) \frac{\ddot{y}_d}{\gamma_1} \\ \left(\frac{1}{r} - 1\right) \frac{\ddot{z}_d - g}{\gamma_1} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (6-17)$$

Ce vecteur révèle que l'intégrale de l'écart entre la position et la position désirée est non nul en régime permanent pour des valeurs non nominales de la masse ($r \neq 1$), l'écart entre la position du drone et la position désirée ainsi que l'écart sur la vitesse linéaire convergent en revanche vers l'origine. Toutefois, les termes non nuls de E_{RP} peuvent être minimisés en réglant judicieusement le gain γ_1 .

L'objectif de ce qui suit est la recherche des gains γ_1, γ_2 et γ_3 satisfaisant les performances précitées pour tout quadrotor ayant une masse m dans l'intervalle $I_m = [m_1, m_2]$.

Le problème d'étude consiste par conséquent à trouver les gains $(\gamma_1, \gamma_2, \gamma_3)$ assurant la stabilité par rapport à Ω de la matrice $A_{BF}(r, \gamma_1, \gamma_2, \gamma_3)$ pour tout paramètre r dans l'intervalle de variation $[\underline{r}, \bar{r}] := \left[\frac{m_N}{m_2}, \frac{m_N}{m_1}\right]$.

L'application liée à ce travail impose la garantie des performances pour $m_1 = m_N$ et $m_2 = 2m_N$, ces valeurs correspondent à une variation du paramètre r dans un intervalle $[\underline{r}, \bar{r}] = [0.5, 1]$.

La démarche à suivre est décrite par les étapes suivantes :

- Vérifier la stabilité générale pour des paramètres nominaux $(r^0, \gamma_1^0, \gamma_2^0, \gamma_3^0)$
- Calculer l'application gardienne $\nu(A_{BF}(r, \gamma_1, \gamma_2, \gamma_3))$ en considérant la masse nominale du quadrotor ($r = 1$)
- Représenter dans l'espace la fonction $\gamma_1 = f(\gamma_2, \gamma_3)$ correspondant aux points d'annulation de l'application gardienne
- Fixer une valeur maximale de γ_1 minimisant ainsi E_{RP} (cette valeur est décrétementée au fil des itérations jusqu'à obtention d'un triplet $(\gamma_1, \gamma_2, \gamma_3)$ assurant la stabilité généralisée de $A_{BF}(r)$ pour $r \in [\underline{r}, \bar{r}]$)
- Effectuer une coupe dans l'espace $\gamma_1 = f(\gamma_2, \gamma_3)$ suivant γ_1 fixé à l'étape précédente
- Déterminer les zones de stabilité, les quadriller selon γ_2 et γ_3 et y fixer une valeur de (γ_2, γ_3)
- Quantifier le domaine de validité $[\underline{r}, \bar{r}]$ pour lequel les valeurs propres de $A_{BF}(r)$ sont confinées dans Ω pour le choix de $\gamma_1, \gamma_2, \gamma_3$ des étapes précédentes
- Si $\underline{r} > 0.5$ pour tout γ_2 et γ_3 balayant la zone stable, décrétement γ_1 et retourner à la cinquième étape

6.3.3 Vérification de la stabilité pour des paramètres nominaux

La première étape lors de l'analyse de stabilité d'un système incertain est de s'assurer du respect des performances pour le cas nominal.

Nous vérifions la stabilité par rapport au domaine Ω de la matrice A_{BF} pour le cas $r^0 = 1$ correspondant au cas nominal $m = m_N$, et en prenant un triplet de gains $(\gamma_1^0, \gamma_2^0, \gamma_3^0)$ pour lequel les valeurs propres de $A_{BF}(r^0, \gamma_1^0, \gamma_2^0, \gamma_3^0)$ sont à l'intérieur de Ω . Un choix possible est donné au Tableau 6.2

Tableau 6.2: Gains nominaux

Gains	Valeurs
\mathbf{r}_1^0	$diag([1.35 \ 1.35 \ 1.35])$
\mathbf{r}_2^0	$diag([3.7 \ 3.7 \ 3.7])$
\mathbf{r}_3^0	$diag([3 \ 3 \ 3])$

Les valeurs propres de $A_{BF}(r^0, \gamma_1^0, \gamma_2^0, \gamma_3^0)$ sont $\sigma(A) = \{-1.2 \pm 0.9i, -0.6\}$ qui sont localisés dans le domaine Ω . Nous disposons alors d'un système nominal Ω -stable.

6.3.4 Calcul de l'application gardienne

Le domaine Ω illustré à la Figure 6.3 est constitué de l'intersection de trois domaines $\Omega_{\alpha_1}, \Omega_{\alpha_2}$ et Ω_{ξ} représentant respectivement une marge de négativité α_1 , une marge de négativité α_2 et un cône d'amortissement ξ tel que représenté à la Figure 6.1.

Pour ce travail, les valeurs numériques des bornes de l'ensemble Ω sont $\alpha_1 = -0.3$, $\alpha_2 = -4$, et $\xi = 0.4$. Ces valeurs permettent d'avoir des pôles plus rapides que la dynamique de la trajectoire lisse qui sera utilisée dans les simulations. Cette trajectoire passe de sa valeur initiale à sa valeur finale en 5 secondes. Ils doivent également être moins rapides que la dynamique des moteurs du quadrotor dont la constante du temps est de $\tau = 0.05sec$ [36]. Ces valeurs peuvent varier selon les exigences de l'application.

Une application gardienne v_{Ω} du domaine Ω est donnée par :

$$v = v_{\xi} v_{\alpha_1} v_{\alpha_2} \quad (6-18)$$

où :

$$v_{\alpha_1}(A) = \det(A \odot I - \alpha_1 I \odot I) \det(A - \alpha_1 I) \quad (6-19)$$

$$v_{\alpha_2}(A) = \det(-A \odot I + \alpha_2 I \odot I) \det(A - \alpha_2 I) \quad (6-20)$$

$$v_{\xi}(A) = \det(A^2 \odot I + (1 - 2\xi^2)A \odot A) \det(A) \quad (6-21)$$

Nous nous intéressons particulièrement aux points d'annulation de l'application v . En effet, ces points limitent les zones de stabilité généralisée de la matrice A_{BF} . Le calcul et la représentation de ces points sont possibles grâce à la fonction « *contour* » de Matlab. Nous représentons à la Figure 6.4, à la Figure 6.5 et à la Figure 6.6 les points d'annulation respectifs des applications v_{α_1} , v_{α_2} et v_{ξ} dans le plan $(\gamma_2, \gamma_3, \gamma_1)$ pour $r = 1$. Une superposition des graphiques permet l'observation des lieux d'annulation de l'application v , le produit en (6-18) est décomposé pour l'observation de l'annulation de chaque application gardienne séparément.

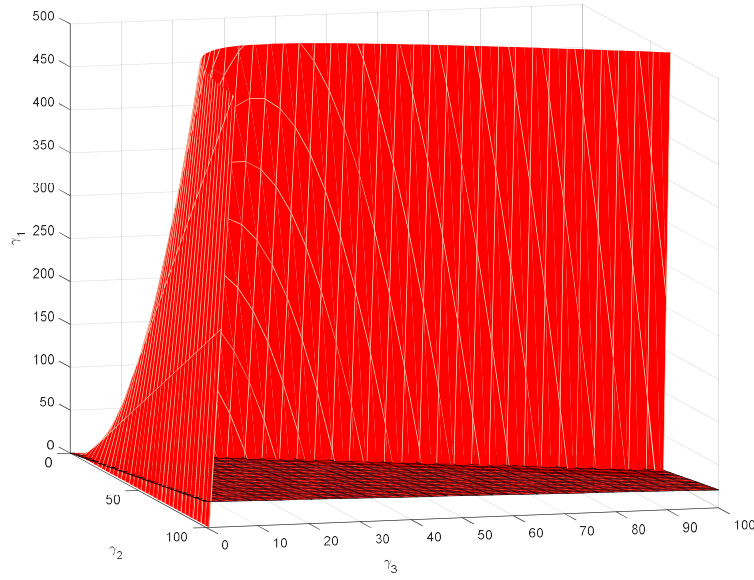


Figure 6.4: Représentation de $v_{\alpha_1} = 0$ dans l'espace $(\gamma_2, \gamma_3, \gamma_1)$

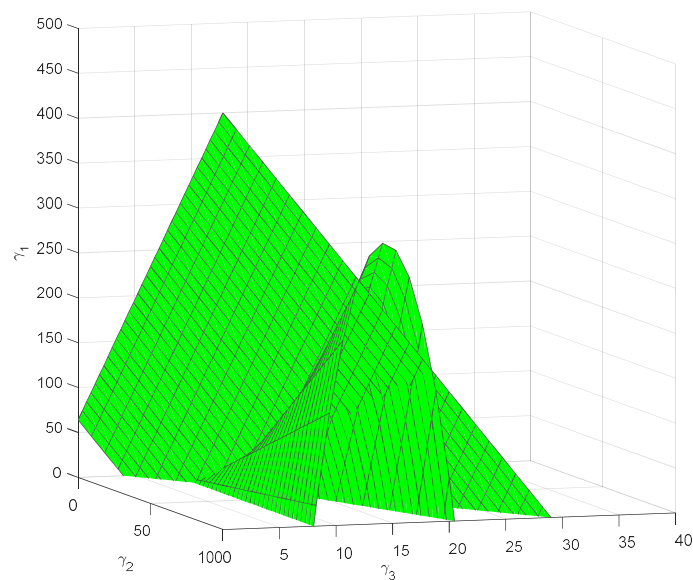


Figure 6.5: Représentation de $v_{\alpha_2} = 0$ dans l'espace $(\gamma_2, \gamma_3, \gamma_1)$

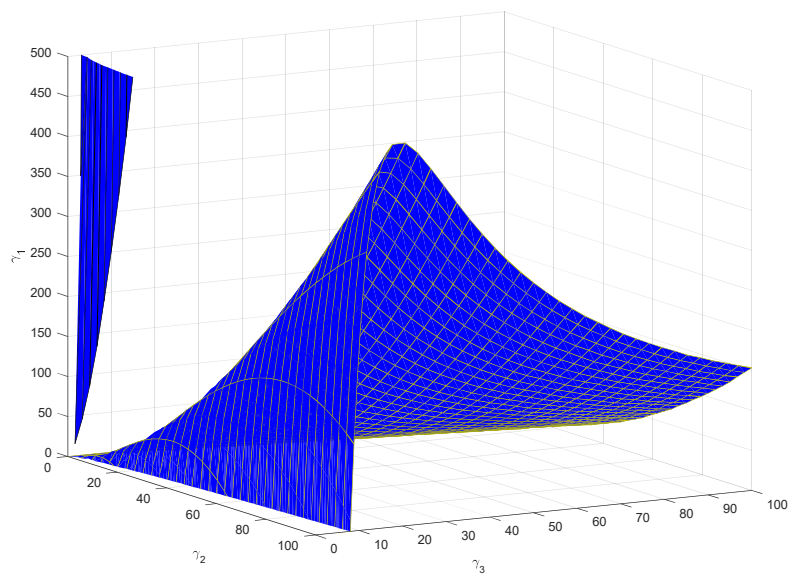


Figure 6.6: Représentation de $v_{\xi} = 0$ dans l'espace $(\gamma_2, \gamma_3, \gamma_1)$

La représentation tridimensionnelle rend difficile le test de stabilité graphique des lieux limités par les points d'annulation. Les composantes stables et instables ne sont pas clairement visibles par conséquent. De ce fait, nous procédons à des découps suivant le gain γ_1 de la Figure 6.7. Cette

action permettra la visualisation, dans le plan 2D, de combinaisons des gains (γ_2, γ_3) en fonction d'un gain γ_1 fixé.

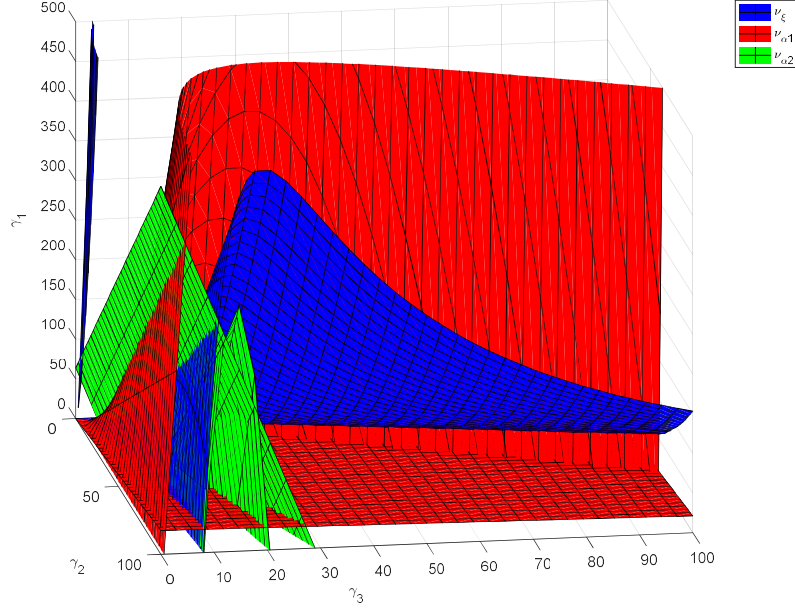


Figure 6.7: Représentation de ν dans l'espace $(\gamma_2, \gamma_3, \gamma_1)$

Dès lors, l'évaluation de la stabilité d'un point à l'intérieur de chaque contour du plan (γ_2, γ_3) reflète la stabilité ou la non-stabilité du contour en entier.

Nous débutons par une illustration de cette méthodologie en procédant à une découpe suivant le contour $\gamma_1 = \gamma_1^1 = 10$. Le résultat est présenté à la Figure 6.8.

Dans le but de déterminer l'ensemble des gains $(\gamma_1^1, \gamma_2, \gamma_3)$, assurant le respect des performances, i.e. le confinement des valeurs propres de A_{BF} dans le domaine Ω , le Tableau 6.3 répertorie un ensemble de points permettant d'évaluer la stabilité ou non de chaque contour. Les valeurs propres de la matrice A_{BF} correspondant à chaque contour sont données à la Figure 6.9.

Sur la base de ces résultats, les gains provenant de la composante C_9 permettent d'obtenir des valeurs propres dans le domaine Ω . C_9 est une composante stable par rapport à Ω . Toute combinaison de gains à l'intérieur de cette zone (voir C_9 à la Figure 6.8) permet de confiner les valeurs propres de A_{BF} dans Ω .

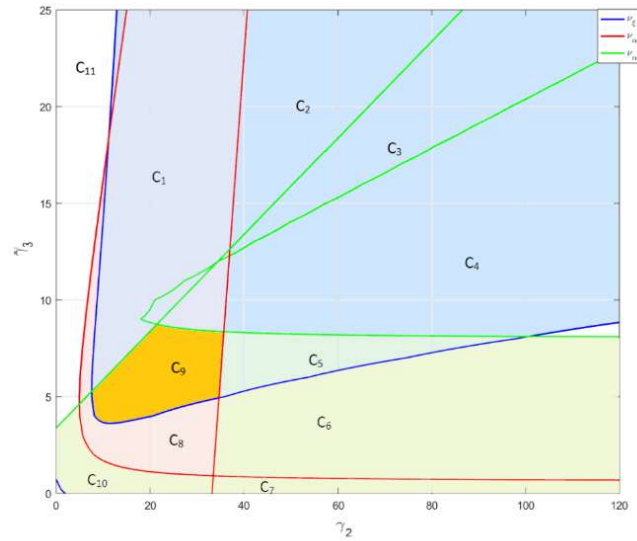


Figure 6.8: Découpe suivant γ_1 et intersection des composantes stables (la zone orange)

Tableau 6.3: Les contours à évaluer

Contour		C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}
Point de test	γ_1	10	10	10	10	10	10	10	10	10	10	10
	γ_2	20	60	80	60	50	40	40	20	20	10	7
	γ_3	15	20	20	10	7	3	1	3	5	1	20

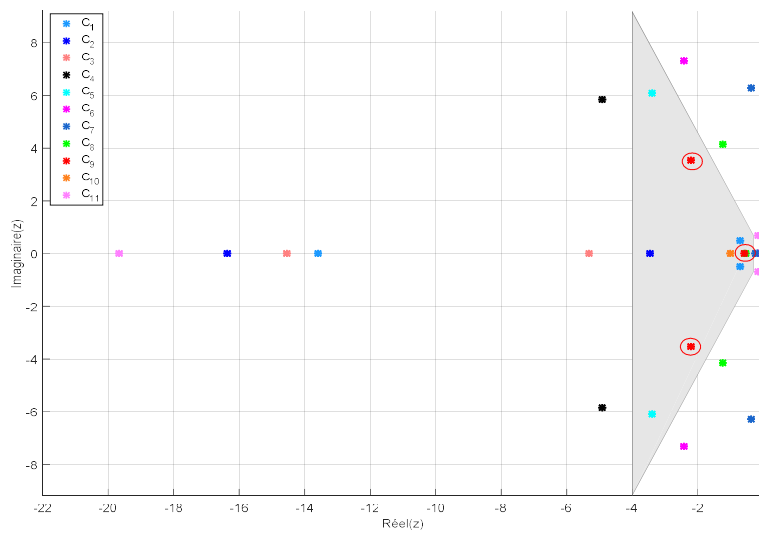


Figure 6.9: Les valeurs propres de A_{BF} selon les contours

À ce stade, la zone de stabilité a été déterminée pour un gain $\gamma_1^1 = 10$. La même procédure est appliquée pour la détermination des zones de stabilité pour d'autres valeurs de γ_1 . Il est intéressant de voir l'évolution de la composante stable suivant les valeurs de γ_1 . La procédure suivie précédemment est réappliquée pour des valeurs de γ_1 entre 1.35 et 200. Le résultat de l'analyse est illustré à la Figure 6.10. Les contours stables y sont mis en évidence par une couleur orange.

On constate ainsi que la frontière d'annulation de l'application gardienne ν_{α_1} balaye le domaine vers la droite lorsque γ_1 augmente, la frontière liée à ν_{α_2} balaye le domaine vers le bas, et enfin la frontière liée à ν_{ξ} vers le haut. Ce comportement aura pour effet le rétrécissement de la zone Ω -stable jusqu'à disparition de zone d'intersection respectant simultanément les trois conditions de stabilité. Ceci produira un gain au-delà duquel la stabilité généralisée n'est plus possible.

Un constat additionnel à relever concerne le gain γ_3 . L'intervalle de validité relié à ce gain varie peu en fonction de γ_1 .

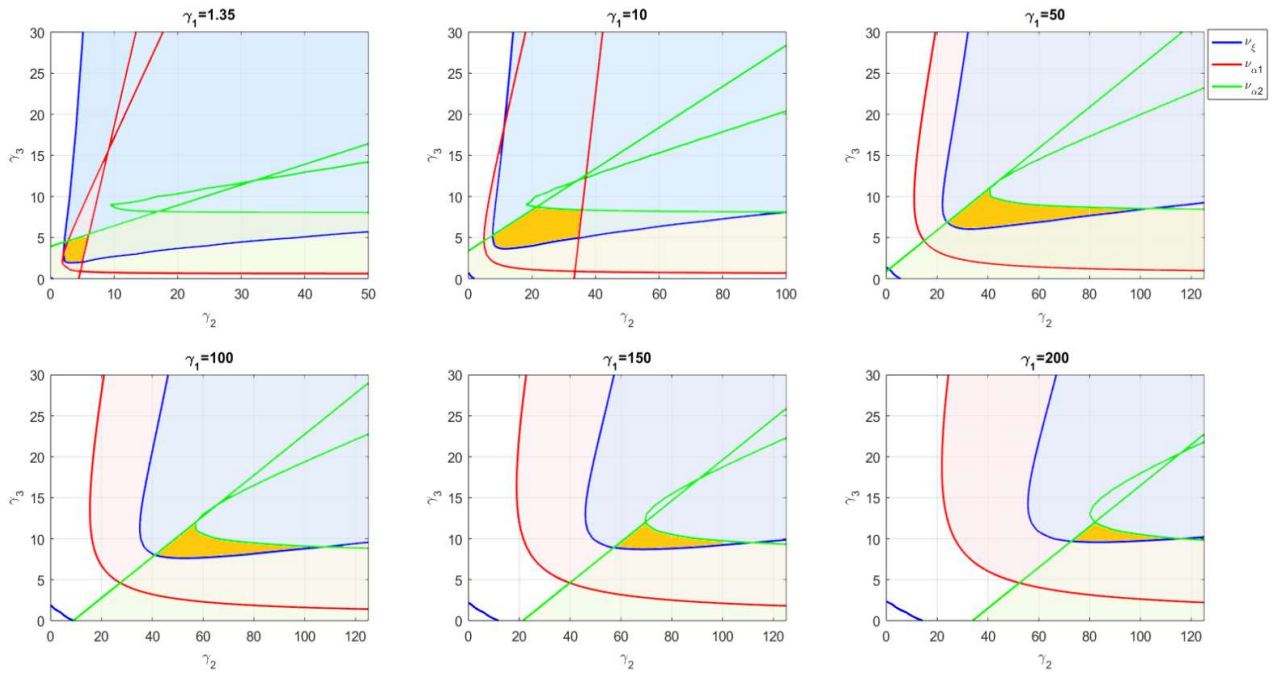


Figure 6.10: Évolution de la composante stable suivant des valeurs de γ_1

6.3.5 Choix des gains du contrôleur

Une étude graphique de la stabilité a permis de trouver un gain maximum $\gamma_1 = 400$ au-delà duquel la stabilité généralisée n'est plus possible, ce constat est confirmé par la Figure 6.11. En effet, le

test de la stabilité des points à l'intérieur de chaque contour n'a permis d'obtenir aucune zone de stabilité dans le cas où $\gamma_1 = 400$. Un seul point à la limite de la stabilité est retrouvé.

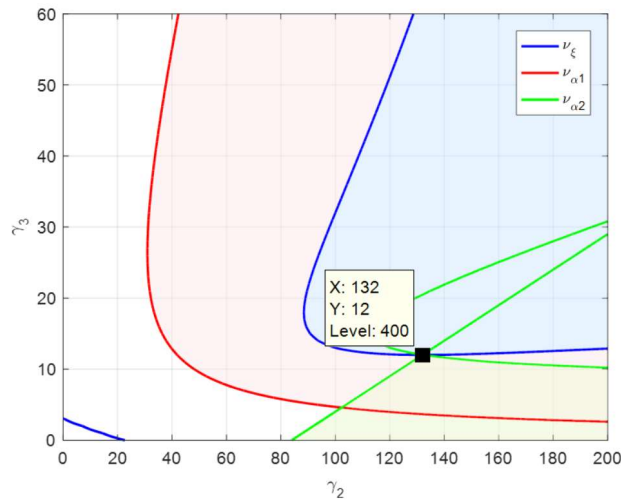


Figure 6.11: Points d'annulation de ν pour $\gamma_1 = 400$

Ce point constitue la valeur maximale que peut prendre γ_1 tout en ayant un domaine non vide de stabilité généralisée.

Nous procédons à présent à l'application des quatre dernières étapes de la stratégie citée en 6.3.2 (p.49). Tout d'abord, une valeur du gain γ_1 est fixé en débutant avec $\gamma_1 = 400$ (la valeur maximale de γ_1). Ensuite, les points d'annulation de l'application gardienne sont tracés similairement à la

Figure 6.8, et l'intervalle de robustesse est examiné sur le domaine de variation des gains γ_2 et γ_3 rattachés à la région stable. Cette séquence est répétée jusqu'à ce que le critère d'itération soit satisfait, soit l'obtention d'un triplet $(\gamma_1, \gamma_2, \gamma_3)$ permettant de garantir les performances exigées pour tout quadrotor dont la masse est dans l'intervalle $[m_N, 2m_N]$.

Un exemple des calculs effectués lors d'une itération est donné pour $\gamma_1 = 200$. Le plan (γ_2, γ_3) est quadrillé pour ce γ_1 et différentes combinaisons de gains sont examinées. Le Tableau 6.4 rapporte un des points du quadrillage à titre d'exemple.

Tableau 6.4: Les gains utilisés lors d'une itération de calcul

γ_1	γ_2	γ_3
200	83	11

Le calcul du domaine de validité $[\underline{r}, \bar{r}]$ est effectué grâce à l'application gardienne $v(A_{BF}(r, \gamma_1, \gamma_2, \gamma_3))$ évaluée aux gains donnés au Tableau 6.4. Le polynôme obtenu est maintenant monoparamétrique.

L'application du Lemme 6.1 est alors possible. Ceci consiste à trouver les solutions de l'équation $v(A_{BF}(r)) = 0$ et de déterminer les valeurs de \underline{r} et \bar{r} correspondant. L'ensemble S des solutions obtenues est :

$$S = \{-8.9e^{-4}, 0, 1.5e^{-4}, 3.5e^{-3}, 0.29, 0.81, 1.1, 1.5, 8.8\} \quad (6-22)$$

Nous concluons que pour cette itération la valeur minimale de l'intervalle de validité est $\underline{r} = 0.81$ et sa valeur maximale est $\bar{r} = 1.1$. Il en résulte que les gains de cette itération permettent de confiner les valeurs propres de A_{BF} dans Ω pour toutes masses dans $[0.9 m_N, 1.23 m_N]$ ce qui ne répond pas à notre objectif. Le passage à une itération suivante est nécessaire.

Pour un γ_1 donné, ces calculs sont répétés pour γ_2 et γ_3 de la composante stable jusqu'à obtention de triplet de gains respectant les spécifications désirées.

Les résultats obtenus sont résumés au Tableau 6.5. Ces valeurs garantissent la stabilité générale sur l'intervalle $[\underline{r}, \bar{r}]$ retrouvé ainsi qu'une erreur minimale de l'intégrale de l'écart de la position en régime permanent.

Tableau 6.5: Valeurs des gains retenus

$(\gamma_1, \gamma_2, \gamma_3)$	$[\underline{r}, \bar{r}]$	$[m_1, m_2]$
(56, 43, 10)	[0.493, 1.014]	$[0.986m_N, 2.028m_N]$

En d'autres termes, nous venons de déterminer une combinaison des gains du contrôleur pour lesquels une variation de la masse dans l'intervalle $I_m = [0.986m_N, 2.028m_N]$ est non nuisible à la qualité du suivi de la trajectoire sans recours à une reconfiguration du contrôleur.

Afin de cerner l'effet du résultat obtenu, une simulation temporelle du comportement du correcteur avec ces gains s'impose. Le véhicule est amené à rejoindre la position d'équilibre partant d'une condition initiale $(x_0, y_0, z_0) = (0, 0, -0.1)$. Nous nous intéressons particulièrement au mouvement vertical qui est le plus sensible à la variation de la masse du quadrotor. Ce scénario est

appliqué à trois quadrotors de masses différentes tel qu'indiqué au Tableau 6.6. Les résultats de simulation sont représentés à la Figure 6.12.

Tableau 6.6: Les modèles de simulation

	Quadrotor 1	Quadrotor 2	Quadrotor 3
<i>masse</i>	m_N	$2m_N$	$3m_N$

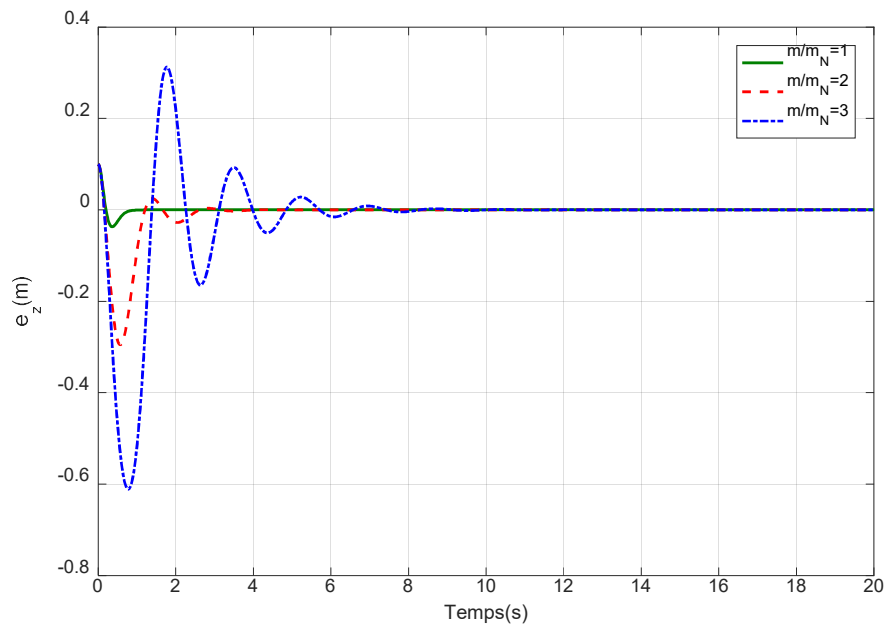


Figure 6.12: Simulation avec les gains retenus

L'erreur e_z représente l'écart entre l'altitude désirée et l'altitude réelle du quadrotor, les oscillations de cet écart s'atténuent et convergent rapidement vers 0 pour le quadrotor 1 et le quadrotor 2. Cette dynamique de convergence correspond aux spécifications du domaine de stabilité Ω . Cependant, l'écart e_z dans le cas du quadrotor 3, dont la masse n'appartient pas au domaine de validité présenté au Tableau 6.5, ne respecte pas la dynamique désirée pour l'erreur. En effet, les pôles du système en boucle fermée ne sont plus dans l'ensemble Ω , ce qui a causé un dépassement plus grand et un amortissement plus lent des oscillations.

6.4 Remarques

Les résultats obtenus jusqu'à maintenant se basent sur une analyse graphique des courbes représentant les points d'annulation des applications gardiennes.

Cependant, la forme de la matrice $A_{BF}(r, \gamma_1, \gamma_2, \gamma_3)$ en (6-14) avec les hypothèses établies en (6-16) permet un calcul analytique de l'intervalle maximale $]\underline{r}, \bar{r}[$ contenant r_0 et assurant la Ω – stabilité.

Dans cette section, nous nous intéressons à la détermination des expressions de \underline{r} et \bar{r} lorsque les gains γ_1, γ_2 et γ_3 sont fixés. La famille de matrices A_{BF} devient alors une famille monoparamétrique $A_{BF}(r)$.

Afin de simplifier la forme des applications gardiennes (6-19) et (6-20), le polynôme caractéristique de A_{BF} est calculé :

$$\det(sI_9 - A_{BF}) = \det(s^3 I_3 + rs^2 \Gamma_3 + rs \Gamma_2 + r \Gamma_1) \quad (6-23)$$

Compte tenu des hypothèses (6-16), le polynôme caractéristique de A_{BF} devient alors :

$$\det(sI_9 - A_{BF}) = (s^3 + r\gamma_3 s^2 + r\gamma_2 s + r\gamma_1)^3 \quad (6-24)$$

Nous constatons que la matrice A_{BF} possède trois valeurs propres σ_1, σ_2 et σ_3 de multiplicité 3. Le placement des valeurs propres de A_{BF} peut alors se ramener au placement des valeurs propres de la matrice:

$$\tilde{A}_{BF}(r) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -r\gamma_1 & -r\gamma_2 & -r\gamma_3 \end{bmatrix} \quad (6-25)$$

Avec cette matrice, nous obtenons :

$$\begin{aligned} \det(\alpha_1 I_3 - \tilde{A}_{BF}) &= (\alpha_1^3 + r\gamma_3 \alpha_1^2 + r\gamma_2 \alpha_1 + r\gamma_1) \\ \det(\alpha_2 I_3 - \tilde{A}_{BF}) &= (\alpha_2^3 + r\gamma_3 \alpha_2^2 + r\gamma_2 \alpha_2 + r\gamma_1) \end{aligned} \quad (6-26)$$

Les applications gardiennes (6-19), (6-20) et (6-21) se calculent alors comme suit :

$$\begin{aligned}
v_{\alpha_1}(\tilde{A}_{BF}) &= -(\alpha_1^3 + r\gamma_3\alpha_1^2 + r\gamma_2\alpha_1 + r\gamma_1) \left(\left(\frac{3\gamma_3^2}{40} - \frac{\gamma_2\gamma_3}{8} \right) r^2 + \left(\frac{\gamma_1}{8} + \frac{3\gamma_2}{40} + \frac{9\gamma_3}{100} \right) r + \frac{27}{1000} \right) \\
v_{\alpha_2}(\tilde{A}_{BF}) &= -(\alpha_2^3 + r\gamma_3\alpha_2^2 + r\gamma_2\alpha_2 + r\gamma_1) \left(\left(-\gamma_3^2 + \frac{\gamma_2\gamma_3}{2} \right) r^2 + \left(16\gamma_3 - \gamma_2 - \frac{\gamma_1}{8} \right) r - 64 \right) \\
v_{\xi}(\tilde{A}_{BF}) &= r^2 \left(\left(-\frac{2\gamma_2\gamma_3^2}{25} + \frac{\gamma_2^2\gamma_3^2}{8} \right) r^2 + \left(-\frac{297}{2500}\gamma_1\gamma_2\gamma_3 - \frac{2\gamma_2^3}{25} \right) r + \frac{729}{125000} \right)
\end{aligned} \tag{6-27}$$

L'application gardienne $v_{\alpha_1}(\tilde{A}_{BF})$ s'annule pour les valeurs suivantes du paramètre incertain r :

$$\begin{aligned}
r_1 &= \frac{27}{1000\gamma_1 - 300\gamma_2 + 90\gamma_3} \\
r_{2,3} &= \frac{25\gamma_1 + 15\gamma_2 - 18\gamma_3 \pm 5\sqrt{(25\gamma_1^2 + 30\gamma_1\gamma_2 - 36\gamma_1\gamma_3 + 9\gamma_2^2)}}{10\gamma_3(5\gamma_2 - 3\gamma_3)}
\end{aligned} \tag{6-28}$$

L'application gardienne $v_{\alpha_2}(\tilde{A}_{BF})$ s'annule pour les valeurs suivantes du paramètre incertain r :

$$\begin{aligned}
r_4 &= \frac{64}{\gamma_1 - 4\gamma_2 + 16\gamma_3} \\
r_{5,6} &= \frac{\gamma_1 + 8\gamma_2 - 128\gamma_3 \pm \sqrt{(\gamma_1^2 + 16\gamma_1\gamma_2 - 256\gamma_1\gamma_3 + 64\gamma_2^2)}}{2\gamma_3(\gamma_2 - 8\gamma_3)}
\end{aligned} \tag{6-29}$$

L'application gardienne $v_{\xi}(\tilde{A}_{BF})$ s'annule pour les valeurs suivantes du paramètre incertain r :

$$\begin{aligned}
r_{7,8} &= 0 \\
r_{9,10} &= \frac{-297\gamma_1\gamma_2\gamma_3 - 200\gamma_2^3 + 36\gamma_1\gamma_3\sqrt{4\gamma_2^2 + 9\gamma_1\gamma_3} \pm 100\gamma_2^2\sqrt{4\gamma_2^2 + 9\gamma_1\gamma_3}}{25\gamma_3^2(-25\gamma_2^2 + 16\gamma_1\gamma_3)}
\end{aligned} \tag{6-30}$$

Pour un triplet $(\gamma_1, \gamma_2, \gamma_3)$ fixé, l'application du Lemme 6.1 permet de déduire le plus grand intervalle $]r, \bar{r}[$ contenant r_0 :

$$\begin{aligned}
r_{min} &= \max_{r_i < r_0} \{r_1, r_2, r_3, r_4, r_5, r_6, r_7, r_8, r_9, r_{10}\} \text{ } (-\infty \text{ sinon}) \\
r_{max} &= \min_{r_i > r_0} \{r_1, r_2, r_3, r_4, r_5, r_6, r_7, r_8, r_9, r_{10}\} \text{ } (+\infty \text{ sinon})
\end{aligned} \tag{6-31}$$

Nous vérifions à cette étape les résultats obtenus pour les valeurs de $(\gamma_1, \gamma_2, \gamma_3)$ choisies auparavant et rapportées au Tableau 6.5, avec ces valeurs nous obtenons :

$$\begin{aligned}
 r_1 &= 6.13 e^{-4} & r_6 &= 1.363 \\
 r_2 &= -2.85 e^{-3} & r_7 &= 0 \\
 r_3 &= 0.204 & r_8 &= 0 \\
 r_4 &= 1.454 & r_9 &= 1.99e^{-3} \\
 r_5 &= 1.014 & r_{10} &= 0.492
 \end{aligned} \tag{6-32}$$

En appliquant (6-31), nous obtenons :

$$\underline{r} = 0.492 ; \bar{r} = 1.014 \tag{6-33}$$

Ainsi, l'intervalle de variation du paramètre r obtenu correspond au résultat de la section 6.3.5

Nous notons qu'il a été possible d'établir des expressions littérales du paramètre r grâce à la forme particulière de la matrice A_{BF} considérée dans ce travail. Toutefois, les étapes décrites auparavant sont valables pour une forme générale de la matrice de la dynamique de l'erreur en boucle fermée.

Ce chapitre a permis de calculer des gains du contrôleur backstepping avec action intégrale respectant le cahier de charge. Ce contrôleur, tel que présenté dans cette section, fera l'objet de validation par simulation dans le chapitre subséquent.

CHAPITRE 7 SIMULATIONS DES CONTRÔLEURS ET VALIDATION

La conception des contrôleurs étant terminée, nous consacrons ce chapitre à la présentation des résultats de simulation.

Le modèle de simulation est présenté dans un premier temps, ensuite une série de simulation des différents contrôleurs est effectuée. Nous testerons la capacité du quadrotor à effectuer diverses manœuvres. Pour les trois contrôleurs présentés précédemment, le suivi de la trajectoire est mis à l'épreuve, d'abord pour des paramètres nominaux dans un environnement parfait, ensuite en présence d'incertitude au niveau de la masse et enfin face aux perturbations dues au vent. À la fin du chapitre, l'efficacité des contrôleurs est évaluée et comparée en se basant sur la moyenne quadratique de l'écart entre les sorties et leurs références.

7.1 Modèle de simulation

L'architecture du modèle de simulation sous Matlab/Simulink est composée de trois blocs essentiels. Un bloc représente la dynamique du quadrotor, un second est consacré au contrôleur à tester et un dernier bloc est responsable de générer la trajectoire à suivre $[x_d, y_d, z_d, \psi_d]^T$. Une quatrième partie modélisant le vent est éventuellement intégrée au modèle lorsqu'il s'agira d'examiner l'effet du vent sur le suivi de la trajectoire par le quadrotor. La structure du modèle est illustrée à la Figure 7.1 et le modèle de simulation correspondant à la Figure 7.2.

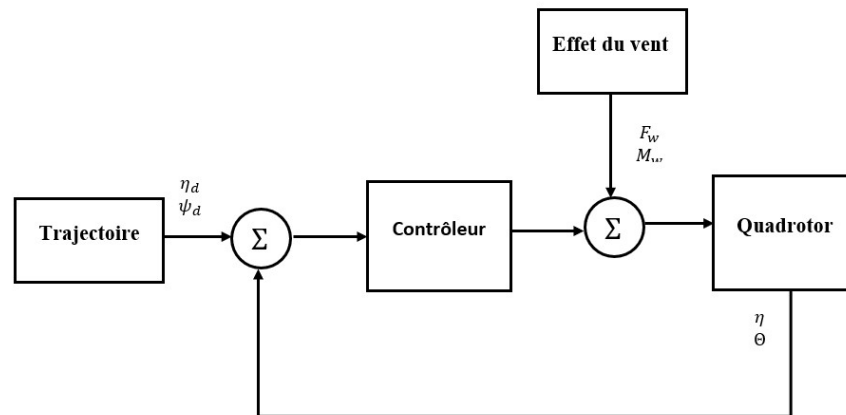


Figure 7.1: Structure du système de simulation

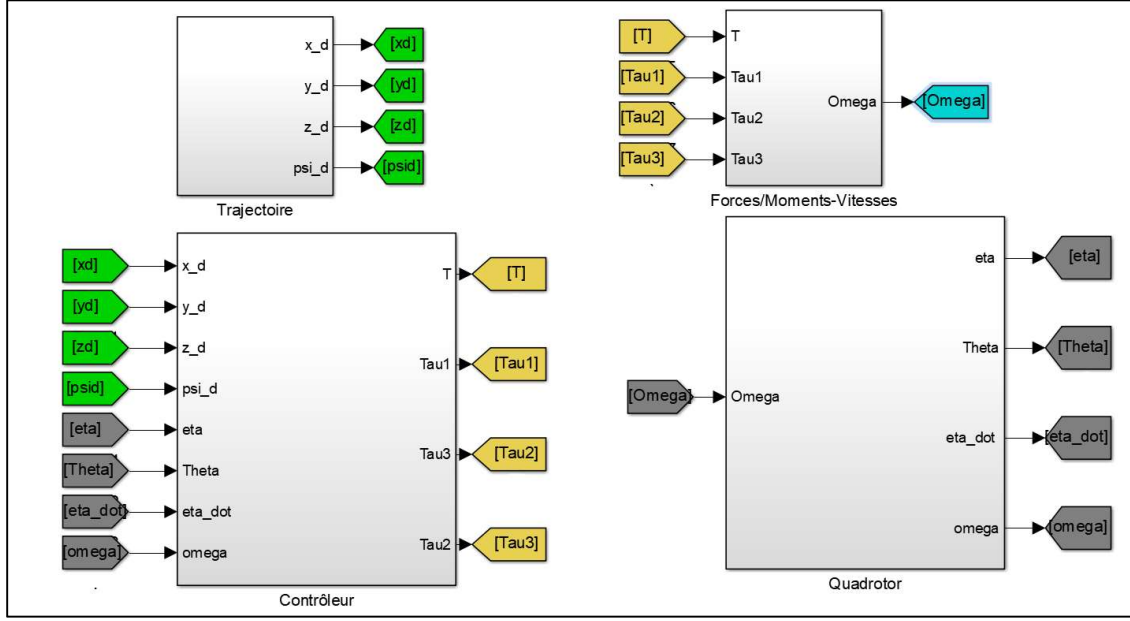


Figure 7.2: Modèle de simulation sous Matlab/Simulink

7.1.1 Modèle du vent

Le modèle du quadrotor établi dans les sections précédentes n'inclut pas l'effet du vent dans l'ensemble des forces agissant sur le véhicule. Cet effet est considéré comme une perturbation externe au système. En simulation, nous désirons évaluer aussi l'action du contrôleur face aux perturbations dues à la présence du vent. Pour cette raison, nous allons modéliser l'effet du vent sur le quadrotor en nous basant sur [37], la force et le moment aérodynamique s'expriment comme suit:

$$F_{aero} = -K_d(\mu - V_{wind}) \quad (7-1)$$

$$M_{aero} = -K_m(\omega - \omega_{wind}) \quad (7-2)$$

F_{aero} , M_{aero} sont respectivement la force et le moment aérodynamiques, et K_d, K_m représentent les coefficients aérodynamiques. V_{wind} et ω_{wind} sont les vitesses de translation et de rotation du vent dans le repère inertiel.

Dans le but d'obtenir un modèle réaliste, nous considérons que le vent est caractérisé par deux composantes. La première est statique, elle est liée à la direction dominante du vent qui ne variera pas pendant la durée du vol. La deuxième composante est stochastique, elle est caractérisée par une

turbulence possédant une densité spectrale de probabilité (DSP). Cette dernière est obtenue, dans ce chapitre, grâce au modèle Dryden fourni par le toolbox « Aerospace Blockset » de Simulink.

La vitesse du vent résultante s'exprime ainsi :

$$V_{wind} = v_{w_{statique}} + v_{w_{stochastique}} \quad (7-3)$$

À titre d'exemple, les vitesses d'un vent de composante statique égale à 6 m/s suivant l'axe x fournies par le bloc Dryden sont illustrées à la Figure 7.3

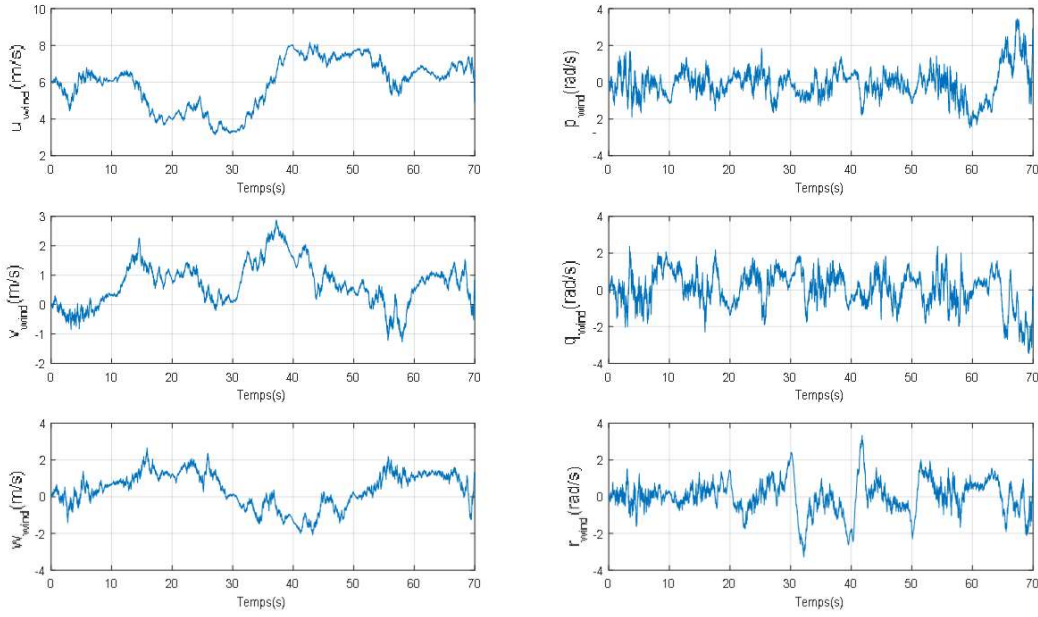


Figure 7.3: Exemple de vent généré en utilisant le modèle de Dryden

7.1.2 Génération de la trajectoire

La technique du backstepping fait intervenir les dérivés de la trajectoire désirée dans l'expression de la loi de commande, à ce titre, il est important de faire suivre le quadrotor une trajectoire suffisamment lisse. Dans ce chapitre, les trajectoires utilisées pour la simulation des contrôleurs par backstepping sont de type parabolique, i.e. à accélération constante. Cette trajectoire commence d'une position q_0 et atteint la position q_f en une durée t . Elle est constituée d'une phase d'accélération pendant T_a et d'une phase de décélération pendant T_d . Nous citons [38] pour plus de détails sur le calcul de la trajectoire. La Figure 7.4 est présentée à titre d'exemple dans le cas où $q_0 = 0$, $q_f = 5$ et $t = 10 \text{ s}$.

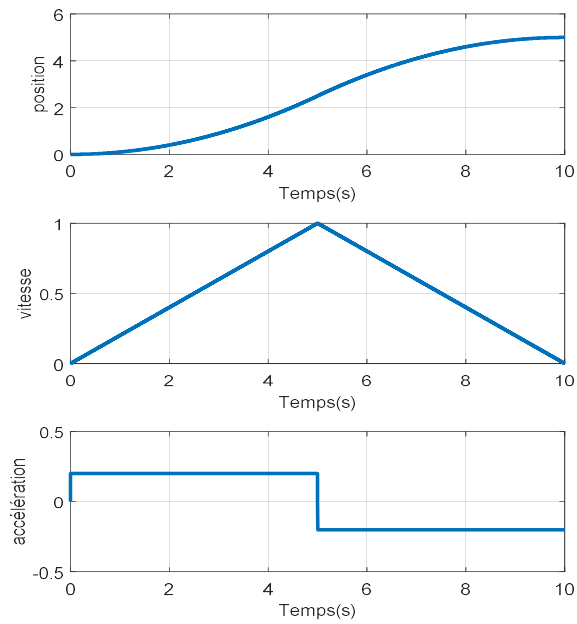


Figure 7.4: Exemple d'une trajectoire à accélération constante

7.1.3 Les paramètres de simulations

Les paramètres utilisés dans la simulation de la dynamique du quadrotor sont ceux d'un Astec Pelican [25], ils sont répertoriés au Tableau 7.1.

Les positions initiales du quadrotor pour toutes les simulations sont :

$$\begin{aligned}
 x_0 &= 0 & \phi_0 &= 0 \\
 y_0 &= 0 & \theta_0 &= 0 \\
 z_0 &= 0 & \psi_0 &= 0
 \end{aligned} \tag{7-4}$$

Tableau 7.1: Paramètres du quadrotor

PARAMÈTRES	SYMBOLES	VALEURS
Masse (kg)	m	1,36
Inertie autour de l'axe x ($kg.m^2$)	I_{xx}	$2,4 \cdot 10^{-2}$
Inertie autour de l'axe y ($kg.m^2$)	I_{yy}	$2,4 \cdot 10^{-2}$
Inertie autour de l'axe z ($kg.m^2$)	I_{zz}	$3,2 \cdot 10^{-2}$

Tableau 7.2: Paramètres du quadrotor (suite)

Longueur de l'hélice (m)	l	0,21
Inertie de l'hélice ($kg.m^2$)	I_h	$7,8187598\ 78 \cdot 10^{-5}$
Constante gravitationnelle (m/s)	g	9,81
Coefficient de poussée ($N.s^2$)	k_t	$1,912444476702 \cdot 10^{-5}$
Coefficient de traînée ($N.m.s^2$)	k_d	$1,84 \cdot 10^{-6}$

7.2 Les scénarios de simulations

7.2.1 Scénario 1: Environnement idéal

Le modèle du quadrotor est considéré parfaitement connu lors de cette simulation. Dde surcroît, nous supposons qu'il ne subit aucune perturbation externe. L'objectif de ce scénario est de valider les performances théoriques relatives au suivi d'une trajectoire donnée. Le quadrotor doit voler à une hauteur h , effectuer une trajectoire rectangulaire, ensuite atterrir à sa position de départ. En parallèle, le mouvement du lacet ψ parcourt une trajectoire désirée ψ_d .

Cette séquence est représentée en simulation par la concaténation d'un ensemble de morceaux. Chacun d'eux est un passage d'un point q_0 à un autre point q_f , le profil de ce morceau est expliqué à la section 7.1.2 . Nous avons fixé une durée $T = 10s$ pour parcourir la distance entre q_0 et q_f , ce temps agit sur l'agressivité de la trajectoire.

La trajectoire désirée pour cette simulation est présentée au Tableau 7.3.

Ayant des objectifs de comparaison, cette trajectoire restera la même pour tous les contrôleurs présentés auparavant.

Tableau 7.3: Points de la trajectoire 1

Points	x_d	y_d	z_d	ψ_d
0	0	0	0	0
1	0	0	-10	$\pi/4$
2	3	0	-10	$\pi/4$
3	3	3	-10	$\pi/4$
4	0	3	-10	$\pi/4$
5	0	0	-10	$\pi/4$
6	0	0	0	0

7.2.2 Scénario 2: Perturbations provenant du vent

Ce scénario est dédié au test de la stabilisation du robot volant en présence de perturbations, notamment celles dues au vent. Nous utilisons pour les simulations le modèle du vent en 7.1.1. Le vent exerce une force sur le quadrotor et introduit un moment aérodynamique. Nous supposons que le vent a la même vitesse statique suivant toutes les directions, qui est égale à $v_{w_{statique}} = 6m/s$. Par la suite nous simulons dans cet environnement la poursuite de la trajectoire décrite au Tableau 7.4.

Tableau 7.4: Points de la trajectoire 2

Points	x_d	y_d	z_d	ψ_d
0	0	0	0	0
1	3	3	-10	0

7.2.3 Scénario 3: Incertitude paramétrique

Ce scénario vise à évaluer la robustesse des contrôleurs conçus face aux incertitudes paramétriques, la masse en l'occurrence.

On reprend un élément de la matrice d'inertie présenté en Annexe A. Les mêmes calculs sont valides pour les autres de la matrice d'inertie.

$$I_{xx} = \int \int \int (y^2 + z^2) \partial m \quad (7-5)$$

Pour une répartition uniforme de la masse, telle que c'est supposé à la section 4.3, cette équation devient :

$$I_{xx} = m \int \int \int (y^2 + z^2) \rho \partial V \quad (7-6)$$

où ρ est la masse volumique du corps. Il s'ensuit que pour une masse $m_\alpha = \alpha m$, le moment d'inertie correspondant à cette variation de la masse est :

$$I_{xx_\alpha} = \alpha I_{xx} \quad (7-7)$$

Sur cette base, le correcteur d'intérêt est bouclé sur un quadrotor ayant une masse m_α et une matrice d'inertie I_α pour chaque simulation. m_α et I_α sont différents de m et I nominaux considérés dans la conception des contrôleurs. Nous avons choisi de varier le rapport $\frac{m_\alpha}{m}$ de 0.5 à 3.5 afin d'analyser l'effet de cette incertitude sur le suivi de la trajectoire.

7.3 Simulation du contrôleur PID

Le réglage des gains du contrôleur linéaire est effectué grâce au toolbox « Simulink Design Optimization ». Des contraintes ont été imposées afin que les sorties x , y et z aient la forme de la réponse d'un système de deuxième ordre standard ayant un temps de réponse à 2% inférieur à $t_{r_{2\%}} = 5s$ et un dépassement maximal $\%OS = 10\%$. Tandis que la sortie ψ est assujetti à suivre la réponse d'un deuxième ordre standard caractérisé par un temps de réponse à 2% inférieur à $t_{r_{2\%}} = 1.5 s$ et un dépassement maximal $\%OS = 10\%$. Ces valeurs correspondent à des valeurs standards dans la littérature que nous utilisons comme base de comparaison.

Les gains obtenus sont rapportés au Tableau 7.5.

Tableau 7.5: Gains du contrôleur PID

	k_p	k_d	k_i
ϕ	2.21	0.15	-
θ	2.21	0.15	-
ψ	0.1526	0.09	-
x	-0.1312	-0.20	-
y	0.1312	0.20	-
z	-3.4	-3.7	-0.9

7.3.1 Scénario 1

Les performances du contrôleur sont évaluées sur le modèle linéarisé ainsi que sur le modèle dynamique non linéaire.

Nous allons effectuer deux essais de vol pour le contrôleur PID. Le premier essai vise à stabiliser le quadrotor autour d'une position désirée $(x_d, y_d, z_d, \psi_d) = \left(3, 3, -10, \frac{\pi}{4}\right)$ qui représente la position autour de laquelle les gains du contrôleur sont réglés. La comparaison entre la réponse du système linéarisé et celle du système non linéaire par rapport à la trajectoire de référence est donnée à la Figure 7.5.

Les réponses temporelles du système linéarisé en x et y ont un dépassement de 0.2% et un temps de réponse à 2% de 3.9 secondes, la réponse temporelle de z se fait avec un dépassement de 0.29% et prend 4 secondes pour atteindre 98% de la réponse finale. Quant à la sortie ψ , elle a un temps de réponse à 2% de 1.4 seconde et dépasse la position désirée de 5.7%. Ces résultats sont résumés au Tableau 7.6. Par conséquent, le contrôleur mis en place permet d'atteindre les performances attendues. À la lumière de la Figure 6.6, nous constatons une détérioration des réponses temporelles du système non linéaire en x , y ainsi que ϕ et θ . Cet effet est dû à la non-linéarité du système et au couplage des états qui ne figurent pas au niveau du modèle linéarisé. Toutefois, les résultats de simulation respectent les spécifications requises pour le temps de réponse et le dépassement maximal.

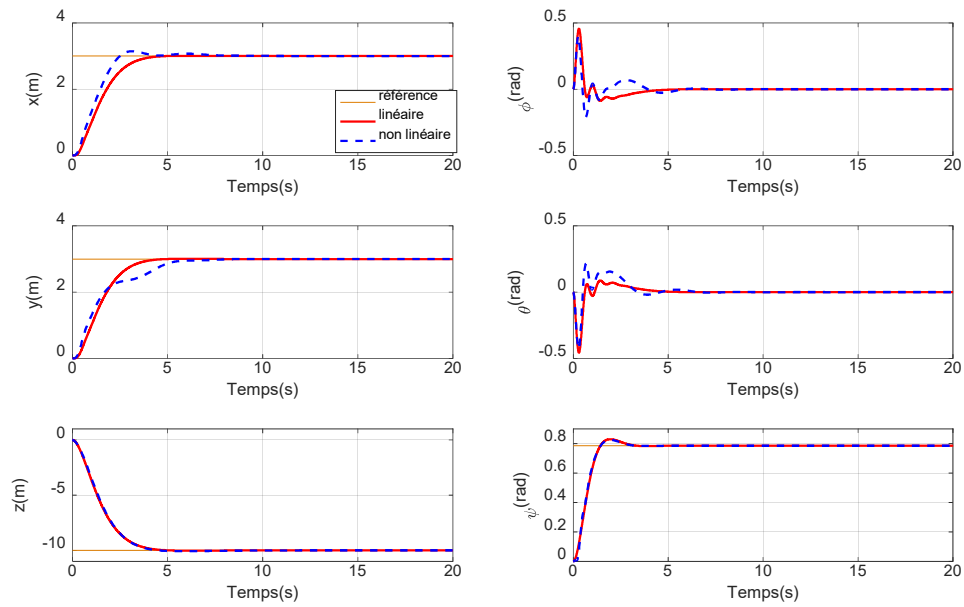


Figure 7.5: Stabilisation autour de (x_d, y_d, z_d, ψ_d) : comparaison entre le modèle linéarisé et le modèle non linéaire

Tableau 7.6: Comparaison des performances

	Critères	Performances	Linéaire	Non linéaire
x	%OS(%)	10	0.2	4.7
	$t_{r_{2\%}}$ (s)	5	39	2.4
y	%OS(%)	10	0.2	0.1
	$t_{r_{2\%}}$ (s)	5	3.9	5.6
z	%OS(%)	10	0.3	0.8
	$t_{r_{2\%}}$ (s)	5	4	3.8
ψ	%OS(%)	10	5.7	4.7
	$t_{r_{2\%}}$ (s)	1.5	1.4	1.36

Le deuxième essai de vol consiste à faire suivre au quadrotor la trajectoire décrite en 7.2.1. La trajectoire effectuée par le quadrotor est visible à la Figure 7.6 ainsi qu'à la Figure 7.7. La Figure 7.8 présente les vitesses de rotation des moteurs qui ne dépassent pas la vitesse maximale tolérée (850 rad/s). Il est clair que le contrôleur PID ne garantit pas un suivi de la trajectoire parfait. En

effet, le contrôleur est conçu autour d'un point de fonctionnement, les performances se dégradent alors plus l'écart de ce point s'accroît.

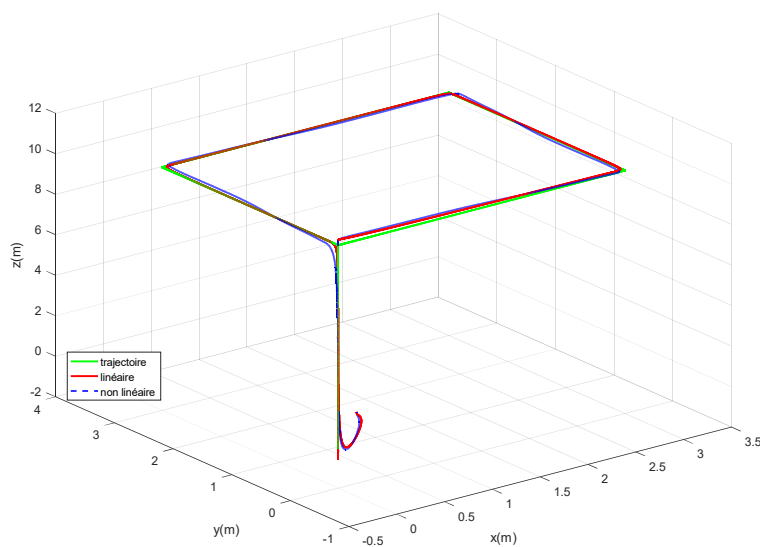


Figure 7.6: Suivi de la trajectoire désirée dans l'espace, cas du PID

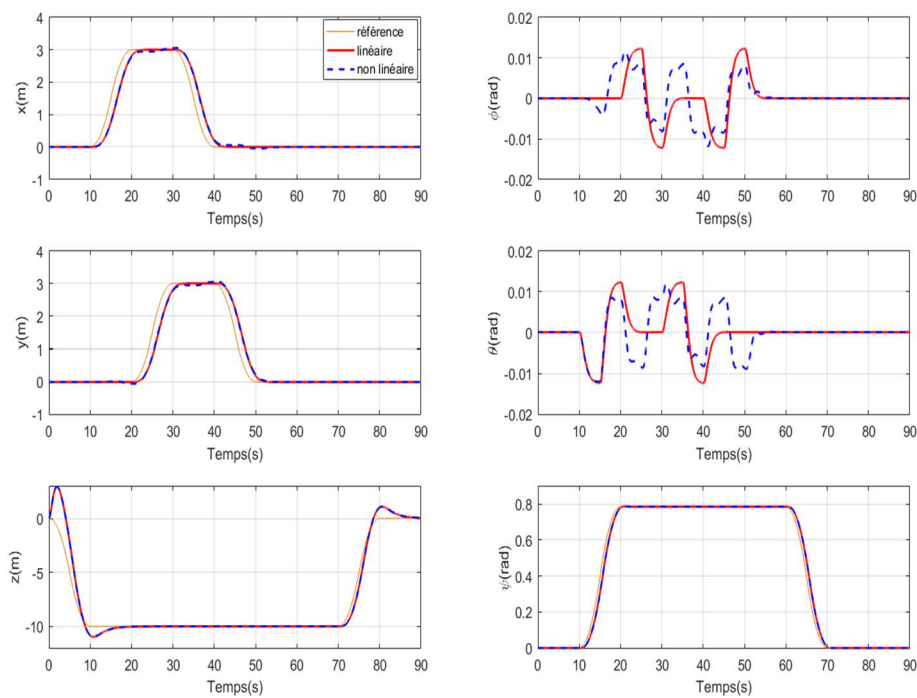


Figure 7.7: Suivi de trajectoires selon les six degrés de liberté, cas du PID

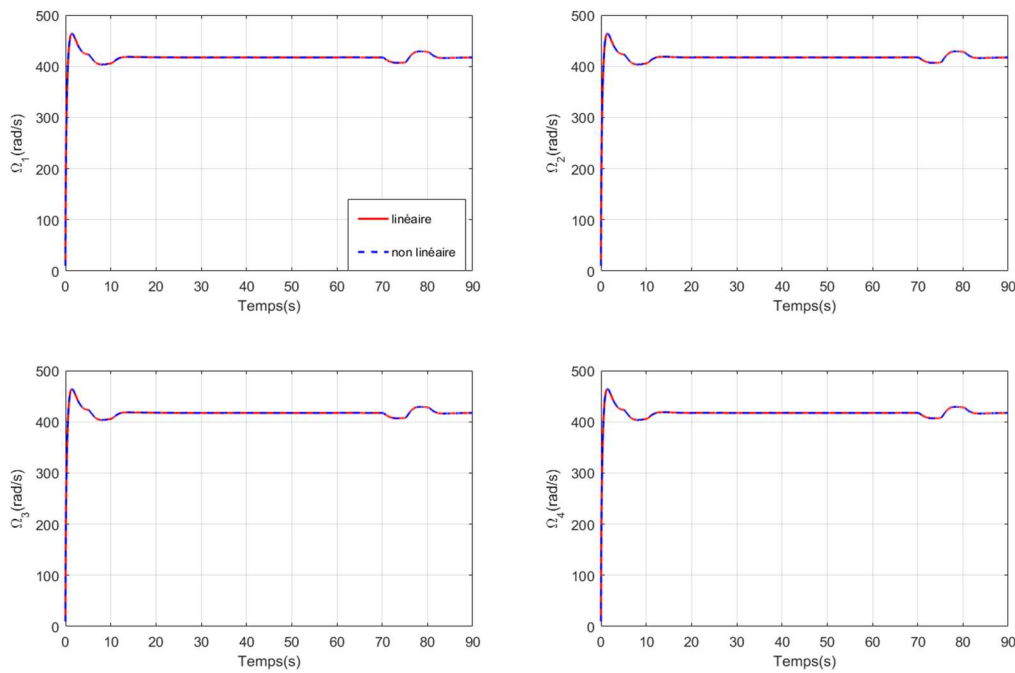


Figure 7.8: Vitesses de rotation des moteurs, cas du PID

7.3.2 Scénario 2

Dans cette partie, le contrôleur est simulé en présence du vent. Le modèle du vent utilisé est développé à la section 7.1.1. Sa force est présentée à la Figure 7.9. On note que la force de poussée du quadrotor pendant un vol stationnaire est égale à $13.8N$. L'effet du vent sur le robot volant ne sera pas alors négligeable. La Figure 7.10 montre les résultats de simulation où on note visuellement l'erreur de suivi en présence du vent, principalement pour le mouvement horizontal suivant les axes x_i et y_i ainsi que le mouvement du lacet. La perturbation du vent est rejetée pour le mouvement vertical grâce à la présence d'une action intégrale au niveau du contrôleur.

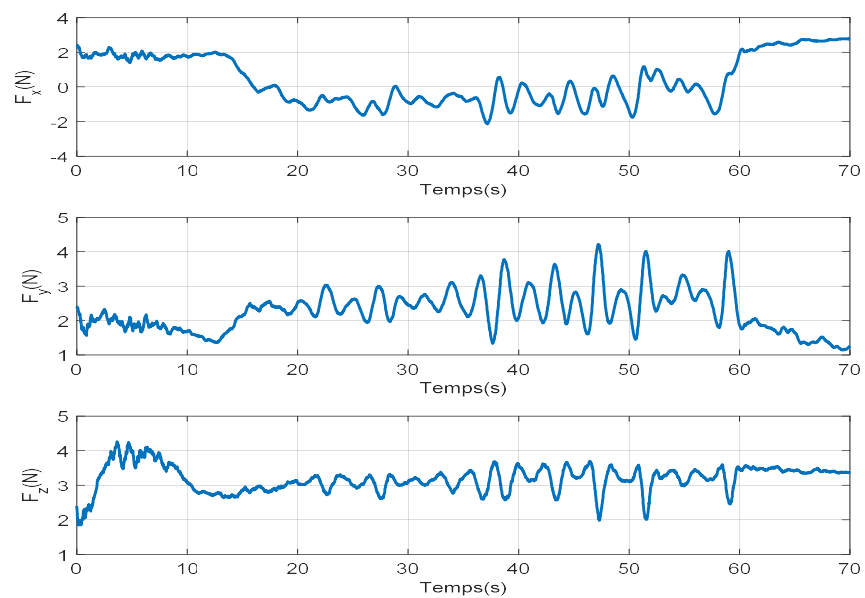


Figure 7.9: La force du vent, cas du PID

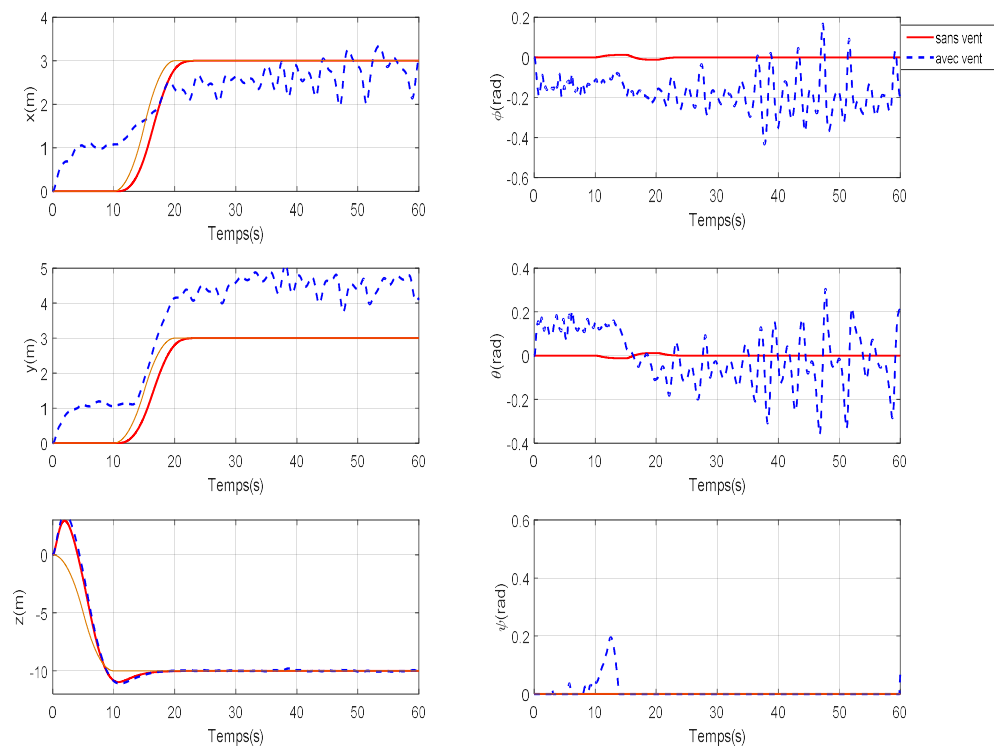


Figure 7.10: Suivi de trajectoires en présence du vent, cas du PID

7.3.3 Scénario 3

Le contrôleur est simulé en cas d'incertitudes paramétriques détaillées à la section 7.2.3. Autrement dit, nous simulons l'implémentation du contrôleur sur des quadrotors ayant la même forme (volume) et des masses différentes. La Figure 7.12 et la Figure 7.11 montrent que la trajectoire horizontale du drone et son mouvement de rotation ne sont pas grandement influés par le changement de sa masse contrairement à la trajectoire verticale. L'écart entre la position désirée et la position réelle selon l'axe z_i devient nulle après l'établissement du régime permanent, cependant l'augmentation du rapport $\frac{m}{m_N}$ provoque une amplification du dépassement en dessus et en dessous de la consigne.

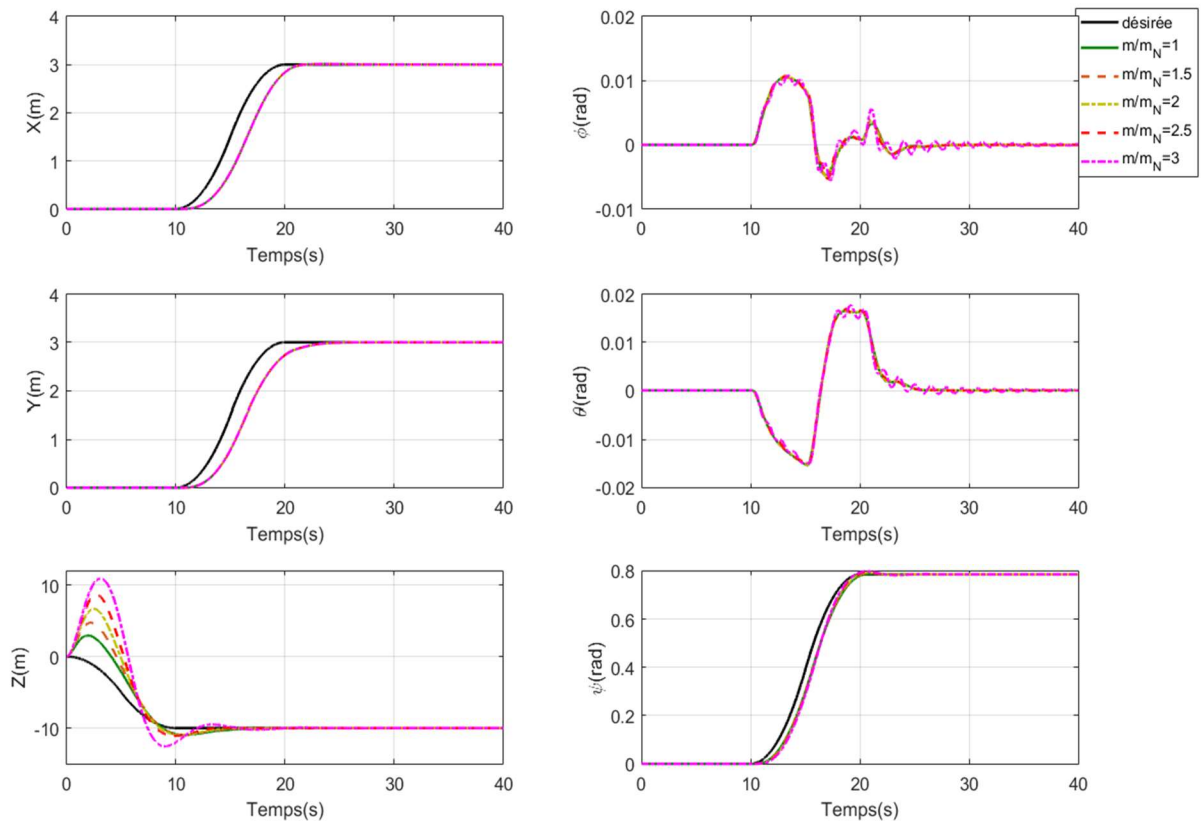


Figure 7.11: Suivi de trajectoires en présence d'incertitudes, cas du PID

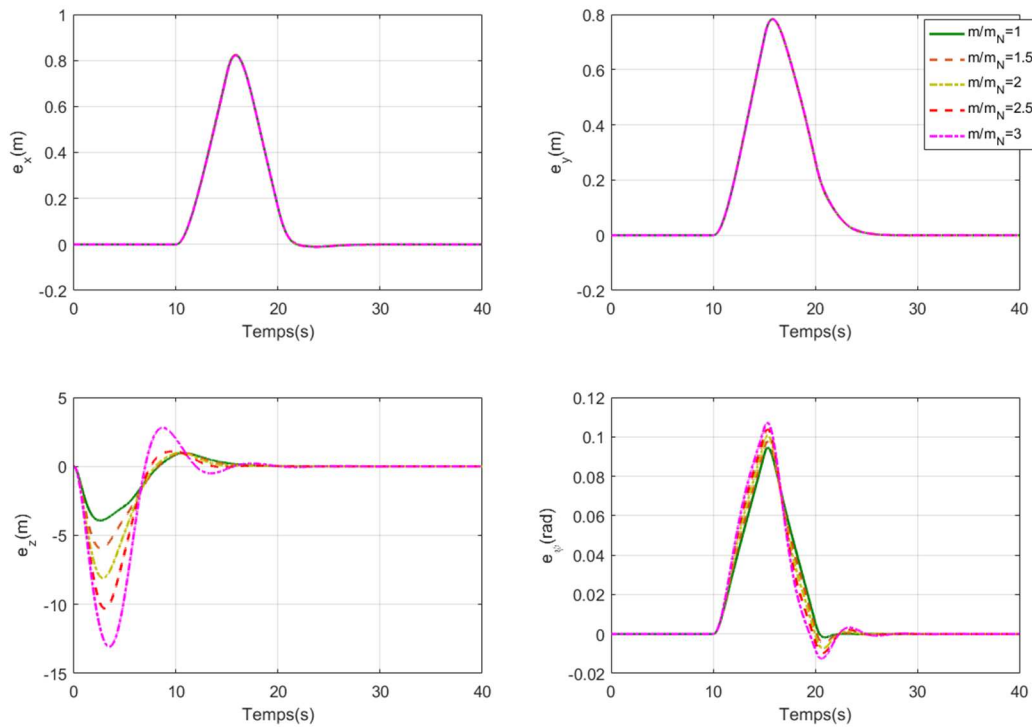


Figure 7.12: Erreurs de suivi en présence d'incertitudes, cas du PID

7.4 Simulation du contrôleur Backstepping

Les gains du contrôleur et des estimateurs utilisés pour la simulation sont présentés au Tableau 7.7.

7.4.1 Scénario 1

Cette simulation consiste à montrer la capacité du drone à suivre des trajectoires sans atteindre les limites des moteurs. Les résultats de la Figure 7.13 valident les performances du contrôleur. Le quadrotor suit parfaitement la trajectoire désirée. Pour plus de visibilité, la Figure 7.14 présente la poursuite de trajectoires selon les six degrés de liberté que possède le quadrotor. Outre la garantie d'une erreur de poursuite nulle, le contrôleur par backstepping génère des entrées de commande qui ne dépassent pas les entrées maximales tolérées par les moteurs du quadrotor (850 rad/s) comme l'illustre la Figure 7.15.

Tableau 7.7: Gains du contrôleur backstepping

Gains	Valeurs
Λ_1	$diag([2 \ 2 \ 2])$
Λ_2	$diag([1 \ 1 \ 0.35])$
Λ_3	$diag([7 \ 7 \ 2])$
Λ_4	$diag([5 \ 5 \ 1.8])$
β_i	5
λ_i	70

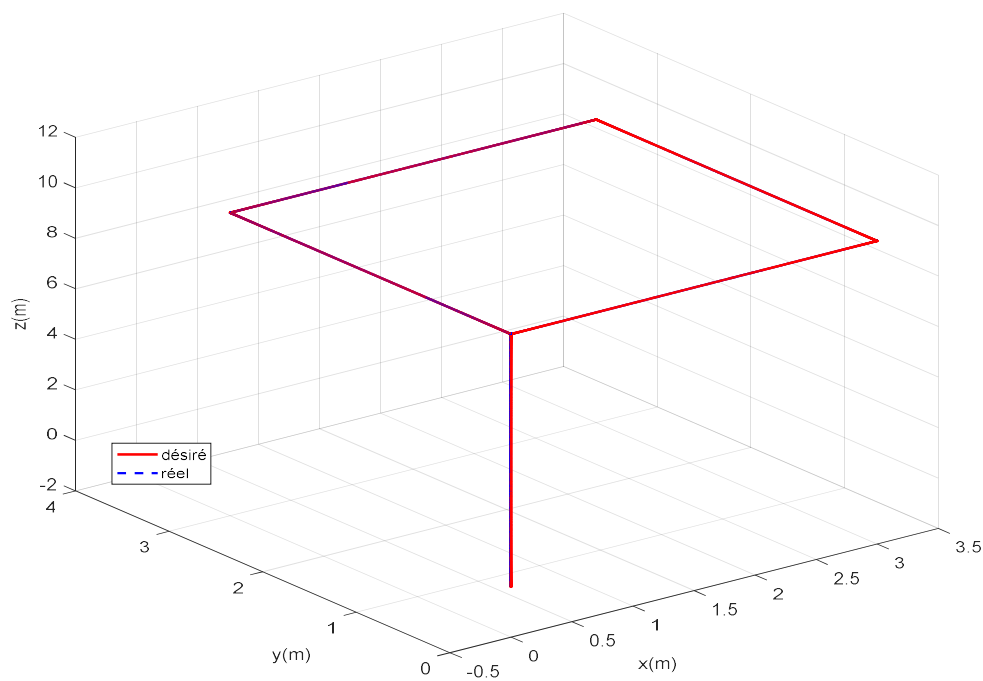


Figure 7.13: Suivi de la trajectoire désirée dans l'espace, cas du backstepping

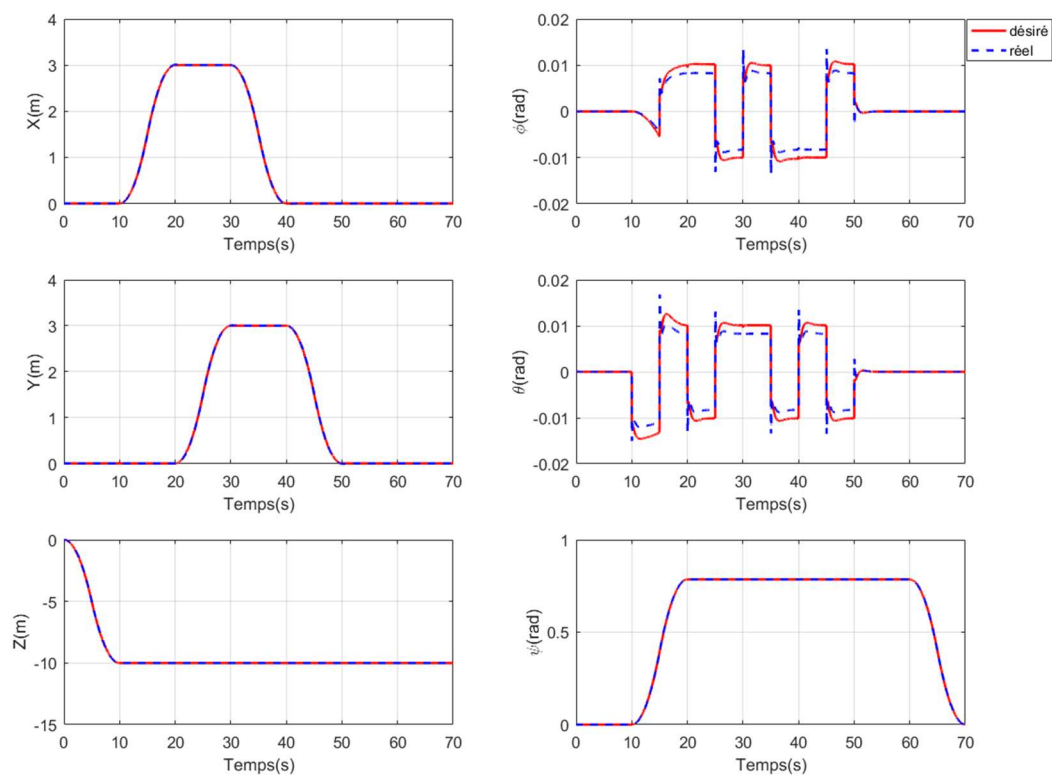


Figure 7.14: Suivi de trajectoires selon les six degrés de liberté, cas du backstepping

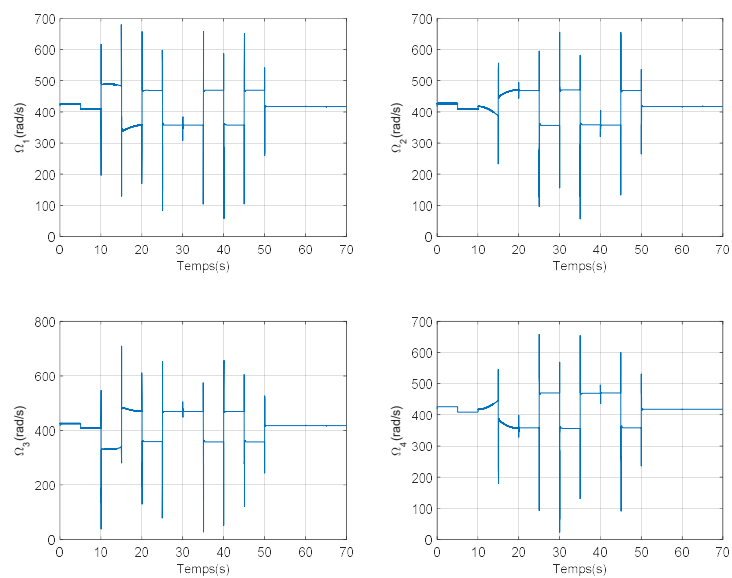


Figure 7.15: Vitesses de rotation des moteurs, cas du backstepping

7.4.2 Scénario 2

La force du vent que subit le quadrotor lors de cette simulation est représentée à la Figure 7.16.

La réponse temporelle du système en boucle fermée par rapport à la trajectoire de référence en présence de cette force est donnée à la Figure 7.17. Sur la base de ces résultats, il est clair que les performances du contrôleur sont dégradées en présence du vent décrit à la section 7.2.2. Nous constatons des erreurs de suivi non nulles. L'erreur maximale est de l'ordre de 17% de la valeur finale pour la position x , de l'ordre de 20% de la valeur finale pour la position y et de 12% pour la position z .

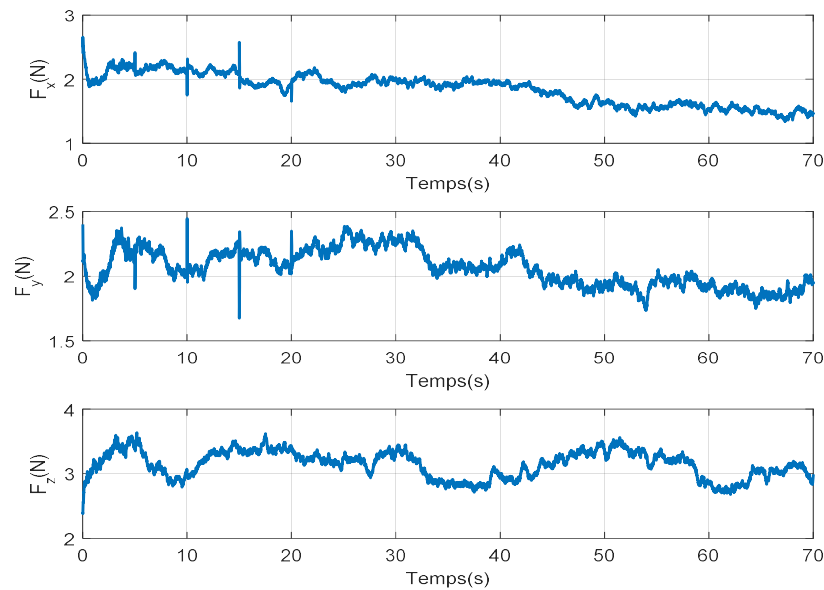


Figure 7.16: La force du vent, cas du backstepping

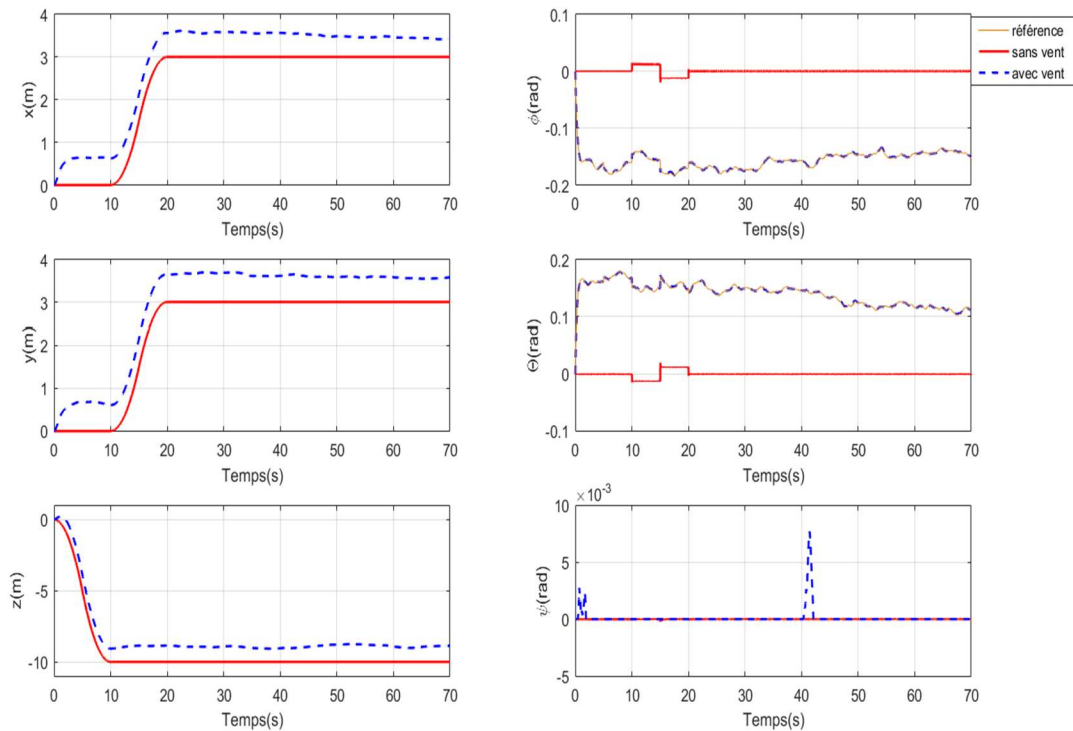


Figure 7.17: Suivi de trajectoires en présence du vent, cas du backstepping

7.4.3 Scénario 3

Ce scénario simule l'incertitude ou la variation de la masse du drone. Les résultats de simulation de la poursuite de trajectoires sont donnés à la Figure 7.18 et la Figure 7.19. Il est évident que les positions longitudinale et latérale ainsi que les angles d'orientation ϕ , θ et ψ sont peu affectées par ce changement. Cependant, la position verticale selon l'axe \mathbf{z}_i est sensible à la variation de ce paramètre. Nous observons un grand écart entre la trajectoire consigne et la trajectoire réelle suivant l'axe \mathbf{z}_i , qui s'amplifie de plus en plus que la masse réelle s'éloigne de la masse nominale pour laquelle la conception est réalisée. Par ailleurs, le contrôleur perd la capacité de générer la force de poussée nécessaire à soulever le corps à l'altitude désirée.

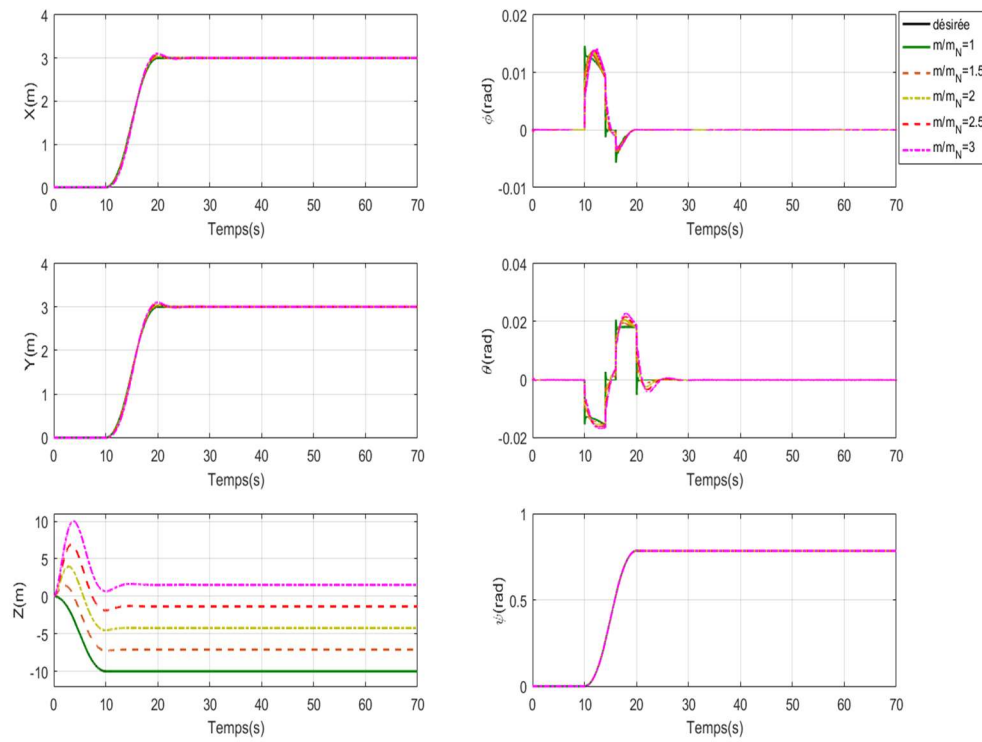


Figure 7.18: Suivi de trajectoires en présence d'incertitudes, cas du backstepping

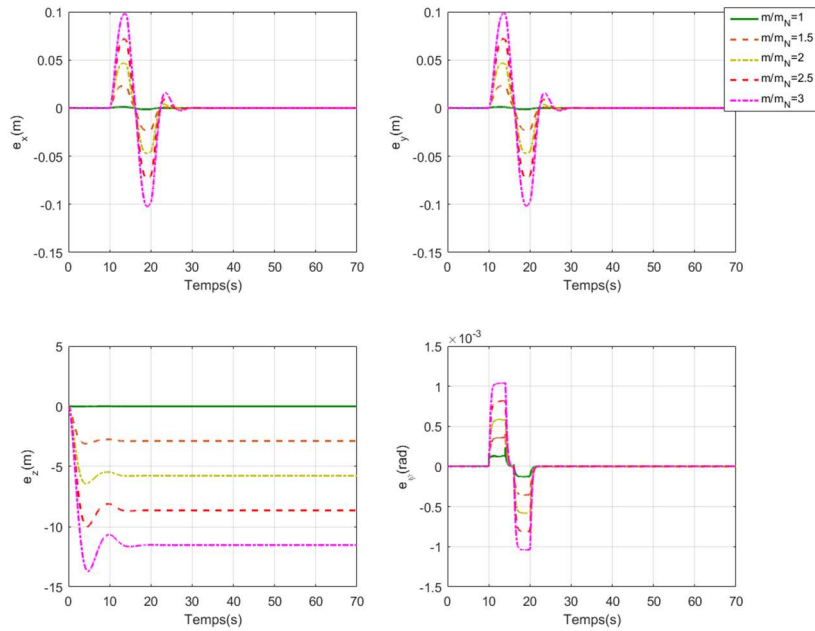


Figure 7.19: Erreurs de suivi en présence d'incertitudes, cas du backstepping

7.5 Simulation du contrôleur backstepping avec action intégrale

Les gains du contrôleur utilisés dans les simulations liées au contrôleur backstepping avec action intégrale sont compilés au Tableau 7.8. Les gains relatifs au contrôleur de la position sont ceux calculés en 6.3.5 par l'outil des applications gardiennes.

Tableau 7.8: Les gains du contrôleur par backstepping avec action intégrale

Gains	Valeurs
Γ_1	$diag([56 \ 56 \ 56])$
Γ_2	$diag([43 \ 43 \ 43])$
Γ_3	$diag([10 \ 10 \ 10])$
K_2	$diag([1 \ 1 \ 0.5])$
A_3	$diag([4 \ 4 \ 2])$
A_4	$diag([1.5 \ 1.5 \ 0.75])$

7.5.1 Scénario 1

Dans cet essai de vol, nous démontrons les performances théoriques du contrôleur par rapport à la poursuite de trajectoires. La Figure 7.20 présente les résultats de la simulation. La trajectoire réelle du quadrotor converge rapidement vers la trajectoire désirée. La Figure 7.21 montre distinctement le suivi de la trajectoire pour chaque degré de liberté. Il est à mentionner que les positions horizontale et verticale du quadrotor ainsi que le lacet convergent vers les positions choisies par l'utilisateur, à savoir (x_d, y_d, z_d, ψ_d) . Pour ce qui est des deux degrés de liberté non actionnés, soit le tangage et le roulis (ϕ, θ) , ils poursuivent les angles ϕ_d et θ_d produits par le contrôleur de la position. Ce contrôleur réalise, théoriquement, l'objectif fixé. La Figure 7.22 prouve également que l'objectif de la commande est accompli sans atteindre les limites des moteurs.

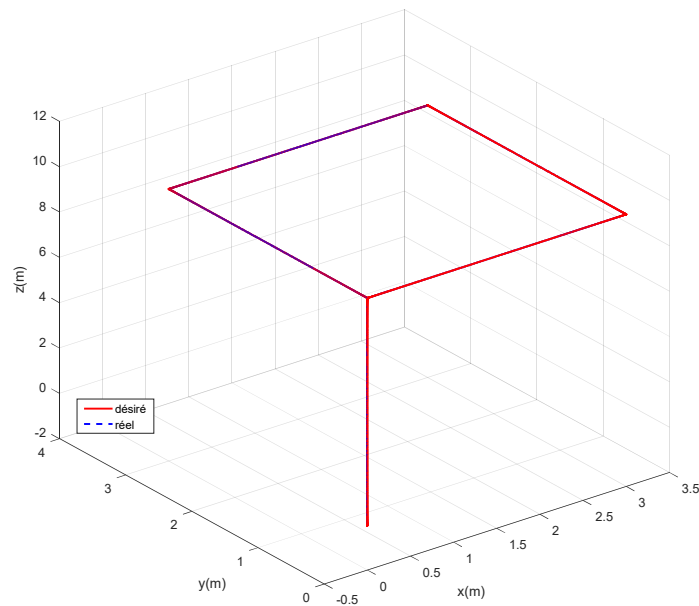


Figure 7.20: Suivi de la trajectoire désirée dans l'espace, cas du backstepping avec action intégrale

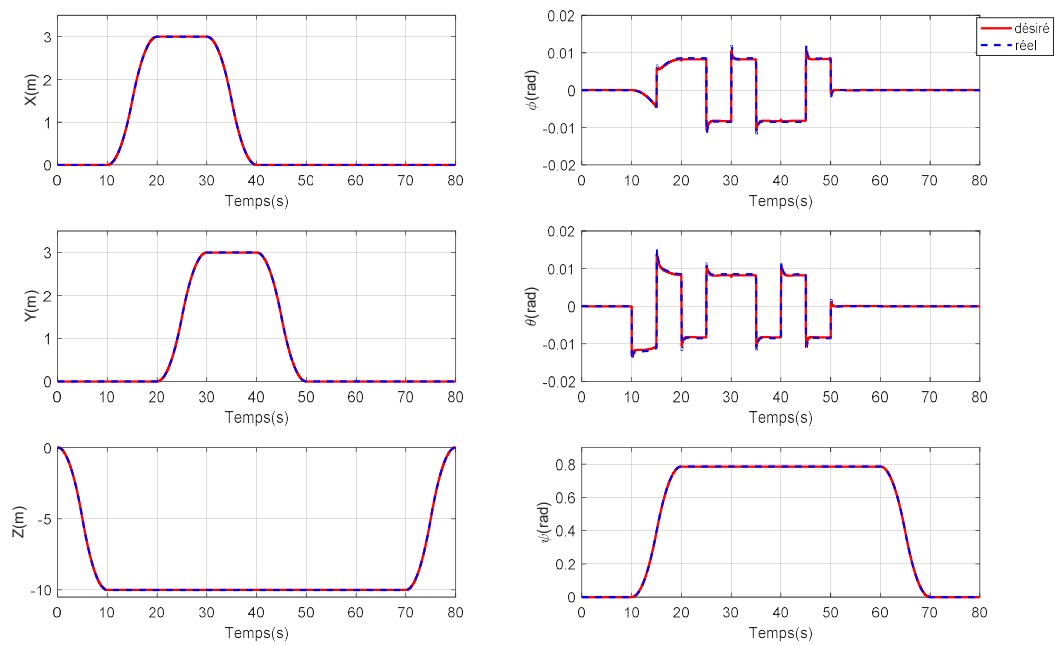


Figure 7.21: Suivi de trajectoires selon les six degrés de liberté, cas du backstepping avec action intégrale

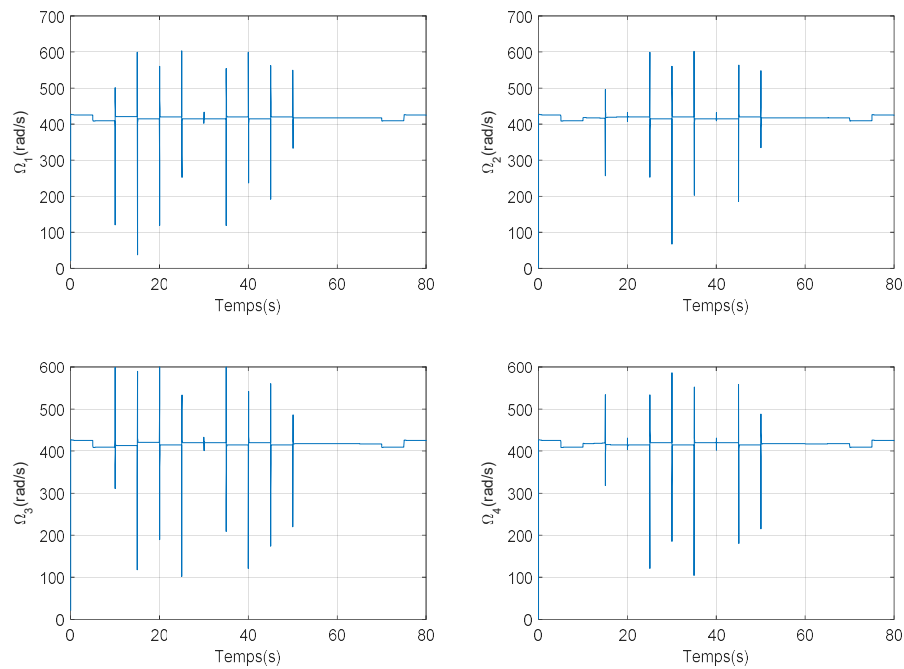


Figure 7.22: Vitesses de rotation des moteurs, cas du backstepping avec action intégrale

7.5.2 Scénario 2

Dans cette partie, nous évaluons le rejet des perturbations externes par le contrôleur backstepping avec action intégrale. La force du vent durant cette simulation est présentée à la Figure 7.23. Sur la base de la Figure 7.24, la réponse du système en boucle fermée par rapport à la poursuite de trajectoires est affectée principalement au début du vol. Après cette transition, l'erreur de suivi de la position devient quasi nulle. Le contrôleur de la position génère des angles d'orientation différentes de ceux obtenus en absence du vent afin de contrer les perturbations qui surviennent, ce qui permet à la trajectoire du quadrotor de converger vers la trajectoire désirée.

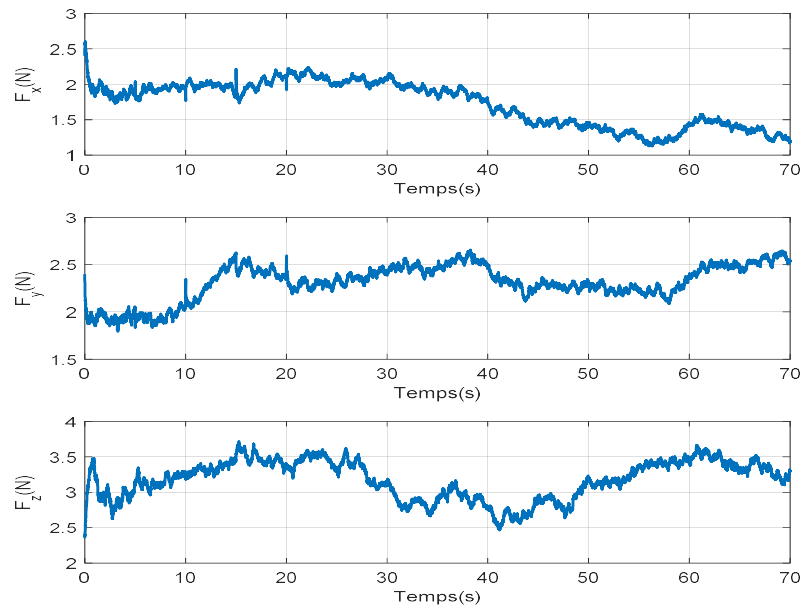


Figure 7.23: La force du vent, cas du backstepping avec action intégrale

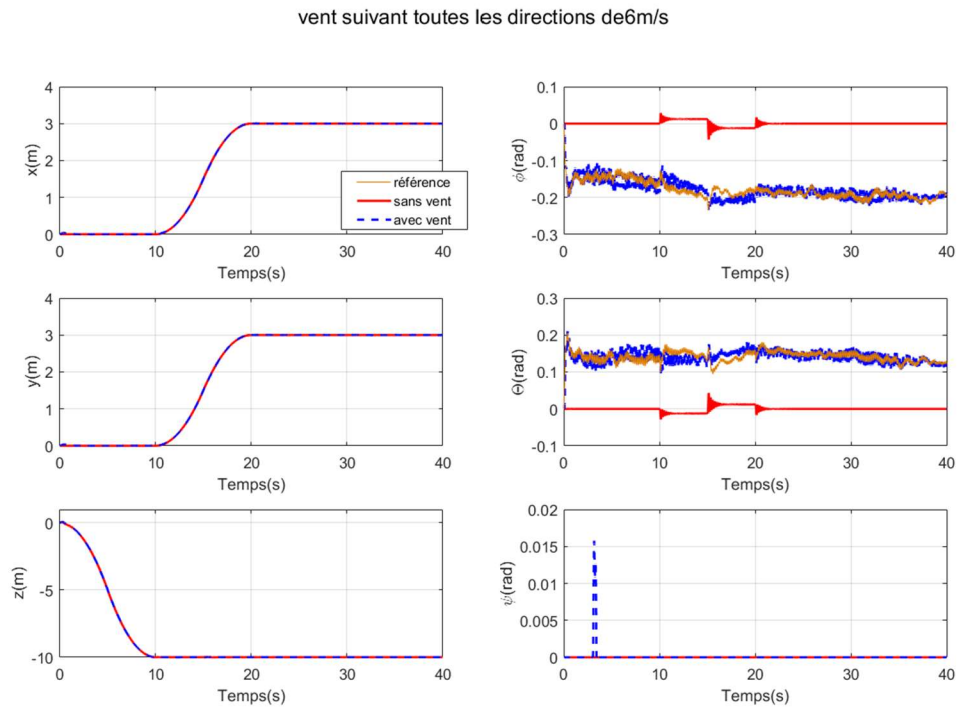


Figure 7.24: Suivi de trajectoires en présence du vent, cas du backstepping avec action intégrale

7.5.3 Scénario 3

Un test de robustesse face aux incertitudes de la masse est effectué dans cette partie. Les résultats de la simulation sont présentés à la Figure 7.25 et à la Figure 7.26. On y remarque que l'erreur de suivi en régime permanent est nulle pour les quatre degrés de liberté actionnés, notamment au niveau de la position z contrairement au résultat retrouvé dans la section 7.4. L'action intégrale annule l'erreur résiduelle. D'autre part, l'erreur de suivi est minimale pour les positions x et y . Quant à la position z , son dépassement au décollage s'élargit de plus en plus que la masse du quadrotor augmente. Toutefois, le suivi de la trajectoire dans les cas d'une masse de $1.5 m_N$ et de $2 m_N$ respectent les performances au niveau de la rapidité de convergence et d'atténuation des oscillations tel qu'attendu. Les gains utilisés pour la simulation ont été calculé pour cet effet.

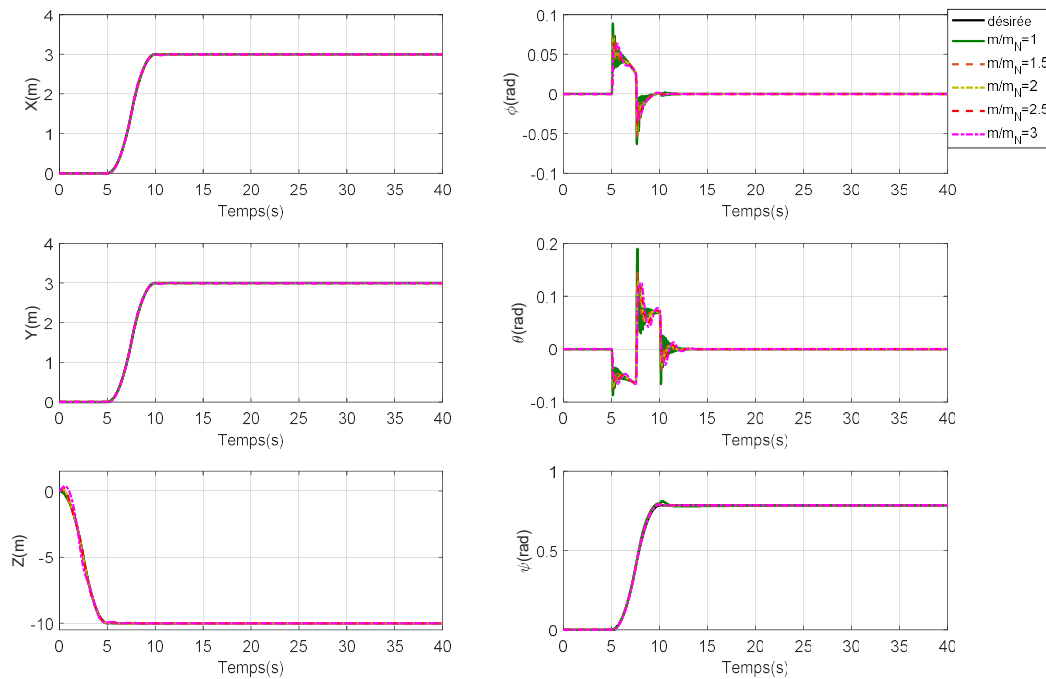


Figure 7.25: Suivi de trajectoires en présence d'incertitudes, cas du backstepping avec action intégrale

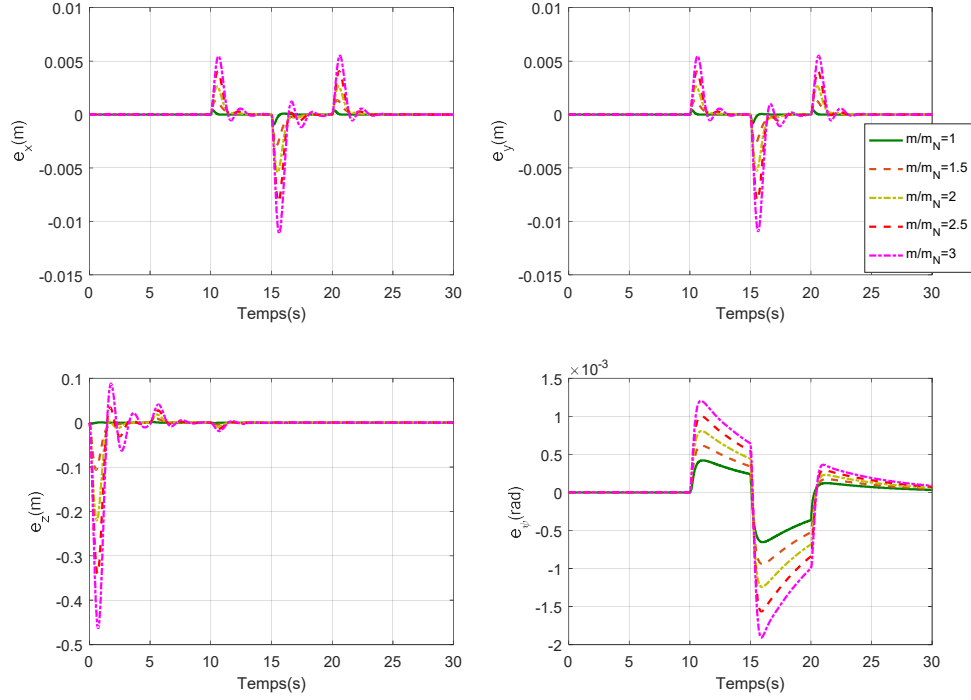


Figure 7.26: Erreurs de suivi en présence d'incertitudes, cas du backstepping avec action intégrale

7.6 Comparaison des résultats

Il existe plusieurs critères de comparaison des performances des contrôleurs, le dépassement de la réponse temporelle et le temps de réponse en l'occurrence. Dans ce paragraphe, nous avons opté pour une étude comparative en nous basant sur la moyenne quadratique de l'écart entre les sorties (x, y, z, ψ) et leurs références (x_d, y_d, z_d, ψ_d) .

Nous comparons les trois contrôleurs dans les trois scénarios de simulation. La première évaluation concerne le suivi de la trajectoire rectangulaire dans un environnement parfait, par la suite nous calculons la moyenne quadratique de l'erreur en présence du vent, et enfin nous nous intéressons à la comparaison en présence d'incertitudes paramétriques représentées par le rapport $\frac{m}{m_N} = 1.5$.

La moyenne quadratique de l'erreur relative à une sortie $s \in (x, y, z, \psi)$ se calcule comme suit :

$$M_q = \sqrt{\frac{1}{n} \sum_{i=1}^{i=n} (s_{d_i} - s_i)^2} \quad (7-8)$$

où s_d est la trajectoire désirée pour la sortie s et n est le nombre de points de la trajectoire.

La valeur de M_q révèle l'écart entre la position réelle du quadrotor et la référence. La sortie concernée est proche davantage de sa trajectoire désirée lorsque cette valeur est petite.

Les moyennes quadratiques obtenues à partir de la simulation des trois contrôleurs dans un environnement idéal avec des paramètres nominaux sont répertoriées au Tableau 7.9. À la lumière de ces résultats, on s'aperçoit que les performances du contrôleur backstepping et du contrôleur backstepping avec action intégrale sont comparables lorsqu'il s'agit de poursuite de trajectoires. En effet, les paramètres utilisés dans la conception des contrôleurs sont identiques aux paramètres du modèle du quadrotor, ce qui engendre une convergence de l'erreur vers l'origine. Étant donné que le contrôleur PID se base principalement sur le modèle linéarisé au voisinage d'un point de fonctionnement, l'erreur de suivi n'est pas négligeable quand le contrôleur est implémenté sur le système non linéaire.

Tableau 7.9: Comparaison dans le cas du scénario idéal

	PID	BS	BSI
$x(m)$	0.2435	0.0005	0.0004
$y(m)$	0.2418	0.0005	0.0003
$z(m)$	0.8885	0.0023	0.0013
$\psi(rad)$	0.0265	0.0000	0.0002

Tableau 7.10: Comparaison en présence du vent

	PID	BS	BSI
$x(m)$	0.7450	0.5427	0.0041
$y(m)$	1.2710	0.6167	0.0039
$z(m)$	1.1056	1.0677	0.0065
$\psi(rad)$	0.1261	0.000	0.0000

Tableau 7.11: Comparaison en présence d'incertitudes

	PID	BS	BSI
$x(m)$	0.1953	0.0070	0.0023
$y(m)$	0.1957	0.0070	0.002
$z(m)$	1.3048	2.8455	0.0187
$\psi(rad)$	0.0217	0.0001	0.0002

Le Tableau 7.10 présente les moyennes quadratiques des écarts du suivi en présence des perturbations dues au vent. Le contrôleur backstepping avec action intégrale présente une meilleure performance concernant le rejet de perturbations, tandis que les autres contrôleurs, le contrôleur PID en particulier, ne parviennent pas à réduire l'erreur lorsqu'une perturbation survient.

La comparaison en présence d'incertitude de la masse est présentée au Tableau 7.11. Dans ce scénario, le contrôleur backstepping avec action intégrale est plus performant. Certes la moyenne de l'écart entre la sortie z et sa référence n'est pas nulle, mais l'erreur en régime permanent l'est. La valeur de cette moyenne résulte principalement d'un dépassement au début du vol (Figure 7.25). L'observation des résultats obtenus avec l'algorithme de commande par backstepping confirme que la position horizontale du robot volant est peu affectée par le changement de la valeur de sa masse contrairement à son altitude pour laquelle la moyenne de l'écart atteint 2,8455 m . L'absence d'une action intégrale au niveau de ce contrôleur ne lui permet pas de produire l'effort de commande nécessaire pour s'opposer à la gravité.

CHAPITRE 8 CONCLUSION ET PERSPECTIVES

La conception d'un système autonome nécessite une certaine expertise dans de nombreux domaines du génie. Toutefois, un seul de ces domaines a constitué le fond de ce mémoire: la commande. L'objectif de ce projet a été accompli. En premier lieu, nous avons pu réaliser une loi de commande non linéaire assurant au quadrotor d'une masse nominale l'accomplissement des missions de vol désirées. Par la suite, le même contrôleur a également permis une qualité de la poursuite de trajectoires satisfaisante lorsqu'implémenté sur une gamme de drones ayant des masses différentes.

La première étape de ce projet concernait le développement des équations dynamiques et cinématiques régissant les mouvements du quadrotor dans l'espace, nous avons élaboré le modèle dynamique grâce au formalisme Newton-Euler puis nous l'avons linéarisé afin d'obtenir un modèle linéaire valable pendant le vol stationnaire. À partir de ce modèle, un contrôleur PID a été d'abord réalisé en vue de constituer une base de comparaison, étant donné qu'il est le contrôleur le plus utilisé dans la commande des drones commercialisés. Ensuite, un contrôleur backstepping avec une action intégrale a été conçu en exploitant la structure en cascade du modèle obtenu. La stabilité de l'ensemble est assurée par l'utilisation des fonctions de Lyapunov. Les contrôleurs ont été validés subséquentement par le biais des simulations. Les trois scénarios de simulation adoptés ont permis de conclure que le contrôleur backstepping avec action intégrale est le plus performant. Non seulement ce dernier a permis d'annuler les écarts entre les positions et les positions désirées causés par les perturbations du vent, mais il a permis également de forcer la convergence de la trajectoire du drone vers la trajectoire souhaitée en présence d'incertitude sur la masse, contrairement au contrôleur backstepping qui ne satisfait pas ces exigences.

Dans un second temps, le contrôleur de l'orientation du quadrotor a été supposé à haute performance dans l'intention de considérer uniquement le contrôleur du sous-système de positionnement dans la synthèse robuste. Ce sous-système s'est présenté en boucle fermée sous forme linéaire en présence d'incertitude sur la masse, ce qui a permis l'appel des applications gardiennes pour le choix des gains assurant le confinement des pôles dans un domaine d'intérêt lorsque la masse du quadrotor varie dans l'intervalle $I_m = [m_N \ 2m_N]$. Les simulations effectuées en tenant compte des gains obtenus ont démontré l'efficacité du contrôleur à accomplir l'objectif fixé.

Bien que le travail présenté dans ce manuscrit ait permis d'atteindre les objectifs initiaux, certaines perspectives mériteraient d'être explorées. L'approche exposée a démontré son efficacité en théorie et en simulation. Cependant, des hypothèses qui ont été faites pour ce projet rendent l'implémentation expérimentale non garantie, mais atteignable. Elles concernent principalement la simplification du modèle de simulation. En effet, le modèle développé ne tient pas compte de la dynamique des moteurs qui est supposée être très rapide. Cependant, ils pourraient être moins réactifs et entraîner un délai pour le changement des vitesses de rotation des hélices telle que requis par la commande. Dès lors, une amélioration majeure concernerait une nouvelle modélisation qui tient compte de cette limitation.

Il est à mentionner qu'une des difficultés rencontrées concerne le calcul des applications gardiennes. Il s'agissait de calculer des déterminants de matrices dont l'ordre est de 36 ce qui amène à des polynômes d'ordre 900. Par conséquent, l'évaluation de l'application sur tous les points d'un maillage se fait lentement sous Matlab. L'utilisation des applications gardienne dans ce travail se basait sur une analyse graphique, il serait intéressant que les gains du contrôleur conservés émanent d'une résolution d'un problème d'optimisation. Ainsi, une fonction pour atténuer l'ordre des applications gardiennes se révèle nécessaire afin d'accélérer la vitesse des calculs.

Finalement, une expérimentation réelle aurait été un atout intéressant pour ce mémoire. Les essais auraient permis de tester la stratégie proposée, d'identifier également les potentiels problèmes émergents et d'envisager des améliorations au contrôleur.

BIBLIOGRAPHIE

- [1] M. Hassanalian and A. Abdelkefi, "Classifications, applications, and design challenges of drones: a review," *Progress in Aerospace Sciences*, vol. 91, pp. 99-131, 2017.
- [2] A. Zulu and S. John, "A review of control algorithms for autonomous quadrotors," *arXiv preprint arXiv:1602.02622*, 2016.
- [3] K. Farid, "Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems," *Journal of Field Robotics*, vol. 29, no. 2, pp. 315-378, 2012.
- [4] L. Li, L. Sun, and J. Jin, "Survey of advances in control algorithms of quadrotor unmanned aerial vehicle," in *International Conference on Communication Technology Proceedings, ICCT*, Hangzhou, China, 2015, vol. 2016-February, pp. 107-111: Institute of Electrical and Electronics Engineers Inc.
- [5] R. Amin, L. Aijun, and S. Shamshirband, "A review of quadrotor UAV: control methodologies and performance evaluation," *International Journal of Automation and Control*, vol. 10, no. 2, pp. 87-103, 2016.
- [6] S. Bouabdallah, A. Noth, and R. Siegwart, "PID vs LQ control techniques applied to an indoor micro quadrotor," in *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, 2004, vol. 3, pp. 2451-2456: IEEE.
- [7] J. Li and Y. Li, "Dynamic analysis and PID control for a quadrotor," in *Mechatronics and Automation (ICMA), 2011 International Conference on*, 2011, pp. 573-578: IEEE.
- [8] I. D. Cowling, O. A. Yakimenko, J. F. Whidborne, and A. K. Cooke, "A prototype of an autonomous controller for a quadrotor UAV," in *European Control Conference (ECC)*, 2007, pp. 4001-4008.
- [9] C. Liu, J. Pan, and Y. Chang, "PID and LQR trajectory tracking control for an unmanned quadrotor helicopter: Experimental studies," in *Chinese Control Conference, CCC*, Chengdu, China, 2016, vol. 2016-August, pp. 10845-10850: IEEE Computer Society.
- [10] K. Zhou, J. C. Doyle, and K. Glover, *Robust and optimal control*. Prentice hall New Jersey, 1996.
- [11] P. Apkarian and D. Noll, "Nonsmooth H_∞ Synthesis," *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 71-86, 2006.
- [12] L. Hao, L. Danjun, Z. Zongyu, and Z. Yisheng, "Robust three-loop trajectory tracking control for quadrotors with multiple uncertainties," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 4, pp. 2263-74, 2016.
- [13] S. L. M. D. Rangajeeva and J. F. Whidborne, "Linear Parameter Varying control of a quadrotor," in *2011 6th International Conference on Industrial and Information Systems, ICIIS 2011 - Conference Proceedings*, Kandy, Sri lanka, 2011, pp. 483-488: IEEE Computer Society.

- [14] O. Araar and N. Aouf, "Full Linear Control of a Quadrotor UAV, LQ vs H," in *2014 UKACC 10th International Conference on Control (CONTROL)*, Piscataway, NJ, USA, 2014, pp. 133-8: IEEE.
- [15] V. Mistler, A. Benallegue, and N. M'sirdi, "Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback," in *Robot and Human Interactive Communication, 2001. Proceedings. 10th IEEE International Workshop on*, 2001, pp. 586-593: IEEE.
- [16] D. Lee, H. J. Kim, and S. Sastry, "Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter," *International Journal of control, Automation and systems*, vol. 7, no. 3, pp. 419-428, 2009.
- [17] H. Voos, "Nonlinear control of a quadrotor micro-UAV using feedback-linearization," in *Mechatronics, 2009. ICM 2009. IEEE International Conference on*, 2009, pp. 1-6: IEEE.
- [18] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro Quadrotor," in *Proceedings - IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005, vol. 2005, pp. 2247-2252: Institute of Electrical and Electronics Engineers Inc.
- [19] F. Munoz, I. Gonzalez-Hernandez, S. Salazar, E. S. Espinoza, and R. Lozano, "Second order sliding mode controllers for altitude control of a quadrotor UAS: Real-time implementation in outdoor environments," (in English), *Neurocomputing*, vol. 233, pp. 61-71, Apr 12 2017.
- [20] F. Chen, R. Jiang, K. Zhang, B. Jiang, and G. Tao, "Robust Backstepping Sliding-Mode Control and Observer-Based Fault Estimation for a Quadrotor UAV," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 8, pp. 5044-5056, 2016.
- [21] D. Lee, T. Ryan, and H. J. Kim, "Autonomous landing of a VTOL UAV on a moving platform using image-based visual servoing," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2012, pp. 971-976: Institute of Electrical and Electronics Engineers Inc.
- [22] H. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, New Jersey: Prentice Hall, 2002.
- [23] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Piscataway, NJ, USA, 2007, pp. 153-8: IEEE.
- [24] T. Madani and A. Benallegue, "Control of a quadrotor mini-helicopter via full state backstepping technique," in *Proceedings of the 45th IEEE Conference on Decision and Control (IEEE Cat. No. 06CH37770)*, Piscataway, NJ, USA, 2006, p. 6: IEEE.
- [25] G. Charland-Arcand, "Contrôle non linéaire par backstepping d'un hélicoptère de type quadrotor pour des applications autonomes " Thesis, École de technologie supérieur, QC, 2014.
- [26] V. Lippiello, F. Ruggiero, and D. Serra, "Emergency landing for a quadrotor in case of a propeller failure: A backstepping approach," in *IEEE International Conference on Intelligent Robots and Systems*, Chicago, IL, United states, 2014, vol. 2014-September, pp. 4782-4788: Institute of Electrical and Electronics Engineers Inc.

- [27] A. R. Benaskeur, *Aspects de l'application du backstepping adaptatif à la commande décentralisée des systèmes non linéaires*. 2002.
- [28] P. Pounds, R. Mahony, J. Gresham, P. Corke, and J. M. Roberts, "Towards dynamically-favourable quad-rotor aerial robots," in *Proceedings of the 2004 Australasian Conference on Robotics & Automation*, 2004: Australian Robotics & Automation Association.
- [29] S. BOUABDALLAH, "Design and control of quadrotors with application to autonomous flying," École polytechnique fédérale de Lausanne, 2007.
- [30] P. Castillo, R. Lozano, and A. E. Dzul, *Modelling and control of mini-flying machines*, 1st ed. Springer-Verlag London, 2005.
- [31] G. Szafranski and R. Czyba, "Different approaches of PID control UAV type quadrotor," *Proceedings of the International Micro Air Vehicles conference*, 2011.
- [32] A. Drouot, "Stratégies de commande pour la navigation autonome d'un drone projectile miniature," Université de Lorraine, 2013.
- [33] L. Saydy, A. L. Tits, and E. H. Abed, "Guardian maps and the generalized stability of parametrized families of matrices and polynomials," (in English), *Mathematics of Control, Signals, and Systems*, vol. 3, no. 4, pp. 345-371, 1990.
- [34] B. R. Barmish and E. Jury, "New tools for robustness of linear systems," *IEEE Transactions on Automatic Control*, vol. 39, no. 12, pp. 2525-2525, 1994.
- [35] D. A. Saussié, "Contrôle du vol longitudinal d'un avion civil avec satisfaction de qualités de manoeuvrabilité," Ecole Polytechnique de Montréal 2010.
- [36] M. Bangura and R. Mahony, "Real-time model predictive control for quadrotors," *19th IFAC World Congress Cape Town, South Africa*, 2014.
- [37] S. Waslander and C. Wang, "Wind disturbance estimation and rejection for quadrotor position control," in *AIAA Infotech@ Aerospace Conference and AIAA Unmanned... Unlimited Conference*, 2009, p. 1983.
- [38] L. Biagiotti and C. Melchiorri, *Trajectory Planning for Automatic Machines and Robots*, 1st ed. Springer-Verlag Berlin Heidelberg, 2008.
- [39] J. M. B. Domingues, "Quadrotor prototype," Instituto superior técnico, 2009.
- [40] M. D. Schmidt, "Simulation and control of a quadrotor unmanned aerial vehicle," University of Kentucky, 2011.

ANNEXE 1 - MOUVEMENT D'UN CORPS RIGIDE À SIX DEGRÉS DE LIBERTÉ

A.1 Matrice de rotation

Le mouvement d'un corps peut être décrit dans une multitude de repères de différentes manières. Dans le cadre de cette étude, nous nous intéressons au mouvement du quadrotor dans le repère inertiel et dans le repère qui lui est lié. D'autre part, la description de sa rotation se fera à l'aide des angles d'Euler qui reste une des méthodes les plus utilisées en aéronautique.

Les trois angles d'Euler sont notés ϕ, θ, ψ . Ils représentent respectivement les angles de rotation autour de x, y et z tel qu'il est décrit à Figure 8.1.

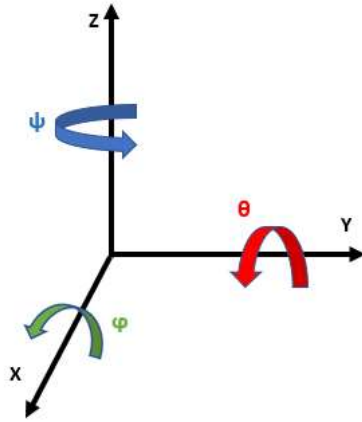


Figure 8.1: Les angles d'Euler

Chaque rotation est décrite par une matrice. Pour les rotations autour de x, y et z , nous avons respectivement les matrices $R_z(\psi), R_y(\theta)$ et $R_x(\phi)$.

$$R_z(\psi) = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (8-1)$$

$$R_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix}$$

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix}$$

La matrice décrivant la rotation du repère lié à l'objet par rapport au repère inertiel est définie par le produit des trois matrices précédentes. Elle est notée $R_{b/i}$.

$$R_{b/i} = R = \begin{bmatrix} \cos(\theta)\cos(\psi) & \sin(\phi)\sin(\theta)\cos(\psi) - \cos(\phi)\sin(\psi) & \cos(\phi)\sin(\theta)\cos(\psi) + \sin(\psi)\sin(\phi) \\ \cos(\theta)\sin(\psi) & \sin(\phi)\sin(\theta)\sin(\psi) + \cos(\phi)\cos(\psi) & \cos(\phi)\sin(\theta)\sin(\psi) - \sin(\psi)\cos(\phi) \\ -\sin(\theta) & \sin(\phi)\cos(\theta) & \cos(\phi)\cos(\theta) \end{bmatrix} \quad (8-2)$$

Cette matrice est orthogonale, c.-à-d. elle est inversible son inverse est sa matrice transposée.

$$(R_{b/i})^{-1} = (R_{b/i})^T = R_{i/b} \quad (8-3)$$

Soit u un vecteur dans l'espace, notons u^i sa représentation dans le repère \mathcal{F}_i et u^b dans \mathcal{F}_b , la relation entre les représentations est alors donnée par l'équation (8-4).

$$u^i = R u^b \quad (8-4)$$

A.2 Matrice de transformation des vitesses angulaires de $\dot{\Theta}$ à ω

La vitesse de rotation ω dans le repère fixe s'exprime en fonction des vitesses de rotation $\dot{\phi}, \dot{\theta}, \dot{\psi}$ dans le repère mobile.

Les angles d'Euler émanent de la séquence de rotation: Roulis (ϕ) \rightarrow Tangage(θ) \rightarrow Lacet(ψ). De ce fait, $\dot{\phi}$ nécessite une rotation, $\dot{\theta}$ nécessite deux rotations et $\dot{\psi}$ résulte de trois rotations successives, par la suite, la vitesse de rotation ω est la somme de trois vecteurs [39].

$$\omega = R(\phi) \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R(\phi)R(\theta) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R(\phi)R(\theta)R(\psi) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \quad (8-5)$$

La relation entre $\dot{\Theta}$ et ω est :

$$\omega = \begin{bmatrix} 1 & 0 & -\sin(\theta) \\ 0 & \cos(\phi) & \sin(\phi)\cos(\theta) \\ 0 & -\sin(\phi) & \cos(\theta)\cos(\phi) \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (8-6)$$

D'autre part, la dérivée des angles d'Euler par rapport au temps s'exprime en (8-7).

$$\dot{\Theta} = \rho(\Theta)\omega \quad (8-7)$$

Avec :

$$\rho(\Theta) = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \frac{\sin(\phi)}{\cos(\theta)} & \frac{\cos(\phi)}{\cos(\theta)} \end{bmatrix} \quad (8-8)$$

Cette matrice est inversible, cependant elle n'est pas orthogonale. Elle met en évidence que l'utilisation des angles d'Euler n'est pas totalement avantageuse. En effet, avec cette représentation ρ présente une singularité lorsque $\theta \in \{(2k + 1)\frac{\pi}{2} / k \in \mathbb{Z}\}$, les deux axes de roulis et de tangage coïncident et un degré de liberté est perdu. Cette situation est expliquée à la Figure 8.2.

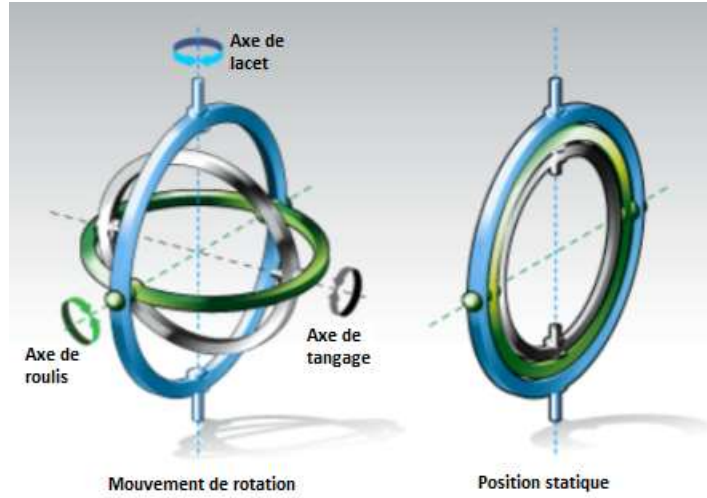


Figure 8.2: Singularité de la matrice ρ , tirée de [40]

A.3 Mouvement d'un corps rigide à 6 degrés de liberté

Nous considérons les repères présentés à la section Chapitre 4. La position du corps est définie par le vecteur allant de O_i à O_b . Ce vecteur est exprimé dans le repère inertiel et est noté $\eta = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$, la

vitesse linéaire exprimée dans le même repère est notée $\dot{\eta} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}$. Quant à l'expression de la vitesse

dans le repère du quadrotor, elle sera notée $V = \begin{bmatrix} u \\ v \\ w \end{bmatrix}$. La vitesse angulaire du repère \mathcal{F}_b par rapport

au repère \mathcal{F}_i est noté $\omega = \begin{bmatrix} p \\ q \\ r \end{bmatrix}$, $\Theta = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}$ pour désigner les angles d'Euler.

L'ensemble des forces et moments appliqués au centre de gravité du corps exprimés dans le repère

lié à l'objet sont respectivement $F = \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}$ et $\tau = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix}$. Le sens positif des moments est défini par

le sens trigonométrique.

À l'aide de la définition de la section (A.1), nous pouvons écrire l'équation cinématique en translation comme suit :

$$\dot{\eta} = RV \quad (8-9)$$

La section (A.2) nous donne l'équation cinématique en rotation :

$$\dot{\Theta} = \rho \omega \quad (8-10)$$

Par application de la deuxième loi de Newton au corps dans le repère inertiel \mathcal{F}_i , en supposant la masse constante, nous obtenons :

$$m \frac{d\dot{\eta}}{dt} = \sum F^i \quad (8-11)$$

Nous projetons cette équation dans le repère \mathcal{F}_b puisque les forces s'y expriment facilement, la formule de Bour appliquée à $\dot{\eta}$ nous donne :

$$\left(\frac{d\dot{\eta}}{dt} \right) = \dot{V} + [\omega]_{\times} V \quad (8-12)$$

Rappelons que l'opérateur $[\]_{\times}$ désigne le produit vectoriel, soit pour deux vecteurs a et b , $[a]_{\times} b = a \times b$.

À partir des équations (8-12) et (8-11), nous obtenons la première équation dynamique en (8-13).

$$\dot{V} = \frac{1}{m}F - [\omega]_{\times}V \quad (8-13)$$

Quant à la deuxième équation dynamique en rotation, elle est obtenue grâce au théorème du moment cinétique appliqué dans le repère inertiel.

Commençons par définir la matrice d'inertie I d'un objet calculé par rapport à son centre de masse.

$$I = \begin{bmatrix} I_{xx} & -I_{xy} & -I_{xz} \\ -I_{yx} & I_{yy} & -I_{yz} \\ -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix}$$

Chaque élément de cette matrice dépend de la distribution de la masse de la manière suivante :

$$\begin{aligned} I_{xx} &= \int \int \int (y^2 + z^2) \partial m \\ I_{xy} &= \int \int \int xy \partial m \end{aligned} \quad (8-14)$$

Les autres termes se calculent de la même manière. Du moment que la masse ne varie pas dans le repère lié au corps rigide, l'expression de I par rapport à ce même repère restera donc constante.

Grâce au théorème du moment cinétique, on obtient :

$$\frac{d(I^i \omega^i)}{dt} = \sum \tau^i \quad (8-15)$$

L'équation de Coriolis pour la dérivée de $I^i \omega^i$ donne :

$$\frac{d(I^i \omega^i)}{dt} = \frac{d(I^b \omega^b)}{dt} + [\omega]_{\times}(I^b \omega^b) = I\dot{\omega} + [\omega]_{\times}(I\omega) \quad (8-16)$$

Comparativement à l'équation dynamique en translation, on exprime (8-15) dans le repère \mathcal{F}_b . En utilisant l'équation (8-16), on obtient le résultat suivant :

$$I\dot{\omega} = \tau - [\omega]_{\times}(I\omega) \quad (8-17)$$